



Universität Hamburg

DER FORSCHUNG | DER LEHRE | DER BILDUNG

Hausarbeit im Seminar:
Softwareentwicklung in der Wissenschaft

**Generative Adversarial Networks in der
Wissenschaft**

Ralf Wedemann

ralf.wedemann@studium.uni-hamburg.de

Studiengang M.Sc. Meteorologie

Matr.-Nr. 6942125

Fachsemester 4

Abgabedatum 31.08.2021

Betreuer: Jakob Lüttgau / Tobias Sebastian Finn

Inhaltsverzeichnis

1	Motivation	1
2	Grundlagen	3
2.1	Convolutional Neuronal Networks	3
2.2	Aktivierungsfunktionen	3
2.3	Aufbau eines Generative Adversarial Network	5
2.4	Klimamodelle und Wolken	5
3	Methodik	7
3.1	Aufbau des verwendeten Generative Adversarial Network	7
3.2	Training	7
4	Ergebnisse	9
4.1	Visuelle Analyse	9
4.2	Spektrale Analyse	10
5	Zusammenfassung und Ausblick	13
	Literaturverzeichnis	15

1 Motivation

Vor dem Hintergrund des Klimawandel spielen Klimamodelle eine wichtige Rolle. Diese können wie ein Tool verstanden werden, um die Ausmaße des Klimawandel vorherzusagen und zu verstehen. So können verschiedene Szenarien berechnet werden für z.B. unterschiedliche Entwicklungen des CO_2 Werts. Auch wenn Klimamodelle viele komplexere Mechanismen, wie z.B. Feedbacks, berücksichtigen, so kann eine Abbildung der Realität nur mit gewissen Einschränkungen modelliert werden. Gründe dafür liegen in der Rechen- und auch Speicherkapazität aktueller Rechnersysteme. Das physikalische Grundprinzip ist jedoch in jedem Klimamodell gleich und bezieht sich auf die Energiebilanz der Erde. Dabei wird die eingehende und ausgehende Strahlung der Erde berücksichtigt, auf die Wolken einen wichtigen Einfluss haben. Vor dem Hintergrund der Rechenkapazität und dem Größenspektrum von Wolken, welches typischerweise zwischen 10^{-6} m und 10^6 m (Schmidt et al., 2020), werden diese meist parametrisiert. In der Parametrisierung wird der Effekt von Wolken aus Messungen abgeschätzt und verwendet ohne diese direkt simulieren zu müssen. Dies ist ebenfalls eine Annahme und wird nur in den seltensten Fällen mit der Realität übereinstimmen.

Generative Adversarial Networks (GAN) haben sich als großen Erfolg im Unsupervised Learning herausgestellt. Diese können verwendet werden um Bilder zu erstellen. Dabei besteht das Netzwerk aus zwei Neuronalen Netzen, die sich gegenseitig beeinflussen und so die Qualität der erzeugten Bilder erhöhen. Anwendungen von GANs finden sich ebenfalls in der Wissenschaft wieder, wie z.B. in der Klimaforschung.

In der Arbeit von Schmidt et al. (2020) wird ein Generative Adversarial Network eingesetzt um Bilder von Wolkenreflexionsfelder zu erzeugen. Diese sollen später benutzt werden um verschiedene Eingangsparameter für Klimamodelle, wie z.B. die optische Dicke, ableiten zu können. Dies kann dazu beitragen Wolken und deren Effekte in Klimamodellen besser berücksichtigen zu können. Das GAN soll mit einem Trainingsdatensatz von meteorologischen Daten trainiert werden, um ein datenbasierten Rahmen zur Reduzierung von Rechenkapazitäten von Klimamodellen zu erschaffen. Die Frage die sich daraus ableitet und untersucht wird ist, wie gut die erstellen Bilder aus dem Modell sind und ob das Vorhaben realistisch ist.

Dazu werden in den folgenden Abschnitten verschiedene Begriffe genauer erläutert und am Ende die Ergebnisse der Arbeit von (Schmidt et al., 2020) vorgestellt.

2 Grundlagen

2.1 Convolutional Neuronal Networks

Convolutional Neuronal Networks, im folgenden als CNN bezeichnet, werden für die Klassifizierung von Bildern eingesetzt. Der Aufbau eines CNN sieht wie folgt aus (Abbildung 2.1). Der Input ist eine zwei bzw. dreidimensionale Matrix, je nachdem ob das Bild nur in Schwarz-weiß vorliegt oder in Farbe. Darauf folgen mehrere Convolution und Pooling Layer. Im letzten Layer, dem fully connected Layer, wird der Output vorbereitet, der sagt wie Wahrscheinlich der Input einer Kategorie wie z.B. Roboter angehört (Convolutional Neural Network, 2021).

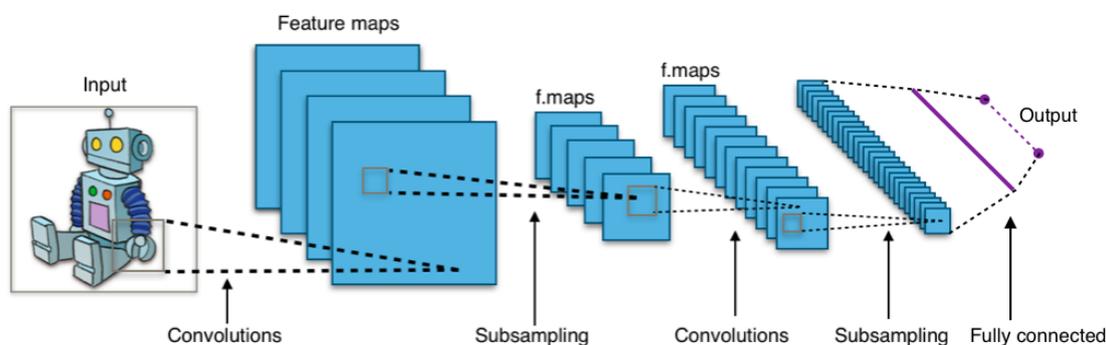


Abbildung 2.1: Aufbau eines typischen CNN zur Klassifizierung eines Bildes.

In den Convolution Layers (dt.: Faltungsschicht) werden mittels verschiedener Faltungsmatrizen Merkmale des Bildes erkannt. Diese Matrizen sind zweidimensional und sind meist 3×3 oder 5×5 groß (Kim, 2017) und können als Filter angesehen werden. Merkmale können z.B. horizontale Linien oder Kreise sein. Das Ergebnis aus der Faltung wird in einer Feature Map, für jedes Merkmal eine, geschrieben und von einer Aktivierungsfunktion (siehe Abschnitt 2.2) verarbeitet.

Der andere angesprochene Layer ist der Pooling Layer. In diesem wird die Größe des Bildes reduziert, indem benachbarte Pixel eines bestimmten Bereiches des Bildes zu einem Wert zusammengefasst werden (Kim, 2017). Dies wird durchgeführt, um nur noch die wichtigsten Informationen des Bildes zu behalten, die für die Klassifizierung benötigt werden. Zwei dieser verwendeten Verfahren sind das Mean Pooling und das Max Pooling, bei denen aus einem 2×2 Bereich der Mittelwert oder das Maximum für die nächste Matrix bestimmt wird (Abbildung 2.2).

2.2 Aktivierungsfunktionen

Die Aktivitätsfunktion (Transferfunktion, Aktivierungsfunktion) stellt den Zusammenhang zwischen dem Netinput und dem Aktivitätslevel eines Neurons dar (Neuronale

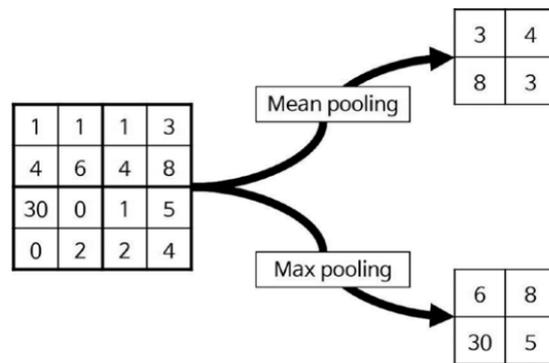


Abbildung 2.2: Graphische Darstellung des 2x2 Mean und Max Pooling (Kim, 2017).

Netze, 2021). Der genannte Netinput ergibt sich durch die Aktivität a_i des vorherigen Neurons und der Gewichtung zwischen des vorherigen und aktuellen Neurons w_i . Sind N Neuronen aus dem vorherigen Layer mit dem betrachteten Neuron verbunden, somit ergibt sich eine Aktivität A von

$$A = \sigma\left(\sum_{i=1}^N w_i a_i\right), \quad (2.1)$$

wobei σ für eine beliebige Aktivierungsfunktion steht. Häufig verwendete Funktionen sind die Sigmoid-Funktion oder die rectified linear unit (ReLU) (Abbildung 2.3). Die Sigmoid Funktion hat den Wertebereich $(0,1)$ und ist eine stetige Funktion. Die ReLU ist bei Klassifikationen die am häufigsten verwendete Aktivierungsfunktion. Ist die Eingabe negativ, so ist die Aktivierung immer Null. Bei positiver Eingabe wird dieser Wert als Ausgabe für die Aktivierung weitergegeben (Ebert, 2019). In einer weiteren Form ist die ReLU etwas abgewandelt worden und gibt bei negativer Eingabe keine Null aus, sondern einen mit einer Konstanten skalierten negativen Wert aus. Diese wird Leaky ReLU genannt.

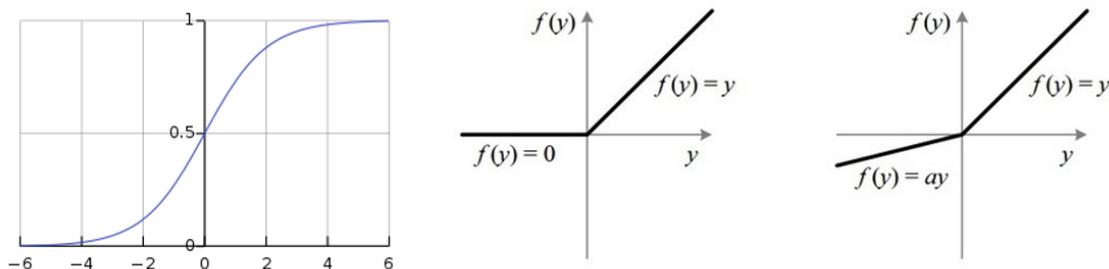


Abbildung 2.3: Darstellung der verschiedenen Aktivierungsfunktionen Sigmoid-Funktion (links), ReLU (mitte) und Leaky ReLU (rechts).

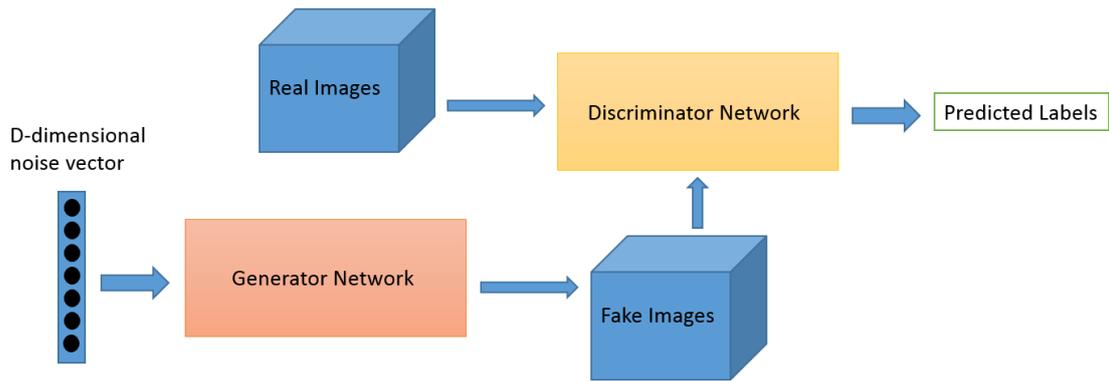


Abbildung 2.4: Typischer Aufbau eines Generative Adversarial Networks.

2.3 Aufbau eines Generative Adversarial Network

In einem Generative Adversarial Network (GAN) stehen sich zwei Netzwerke gegenüber. Zum einen gibt es den Generator, welcher aus einem Noise Vektor ein Bild erstellen kann und zum anderen gibt es den Diskriminator, welcher lernt zu bestimmen, ob ein Bild aus dem Generator oder aus einem Datensatz echter Bilder stammt. Um dies zu Veranschaulichen kann sich der Generator als ein Team von Geldfälschern vorgestellt werden, die versuchen ihr Falschgeld zu verwenden. Der Diskriminator ist analog zur Polizei und versucht das Falschgeld zu erkennen. Der Diskriminator D wird demnach so trainiert, dass die Wahrscheinlichkeit zu erkennen, ob das Bild aus dem Generator oder aus dem Trainingsdatensatz kommt, maximiert wird. Gleichzeitig wird G so trainiert, dass $\log(1 - D(G(z)))$ minimiert wird. Es wird also ein Nullsummenspiel zwischen beiden Netzwerken gespielt (Goodfellow et al., 2014).

2.4 Klimamodelle und Wolken

Klimamodelle sind eines der wichtigsten Werkzeuge, um die Konsequenzen des Klimawandel im Bezug auf z.B. Änderung im Niederschlag zu verstehen und zu vorherzusagen (Schmidt et al., 2020). Diese beruhen auf dem wesentlichen Grundprinzip der Energiebilanz der Erde. Das ist die Bilanz aus der eingehenden und ausgehenden Strahlung am Oberrand der Atmosphäre. In diesem System haben Wolken eine wichtige Rolle. Diese beeinflussen die kurzwellige Strahlung durch Absorbition und Reflektion (Abbildung 2.5).

Da Klimamodelle meist als Domain die komplette Welt abbildet und dies über einen langen Zeitraum ist die Auflösung der Gitter in den Modellen durch die Rechenkapazitäten begrenzt. Wolken haben ein großes Spektrum in ihrer Größe. Von kleinsten Cumulus Wolken bis hin zu geschlossenen Wolkendecken bei Tiefdruckgebieten. Die direkte Berechnung von kleinen Wolken ist nicht möglich bzw. wäre sehr Rechenaufwendig. Aus diesem Grund sind Wolken meist parametrisiert, was jedoch dazu führt Annahmen zu

treffen.

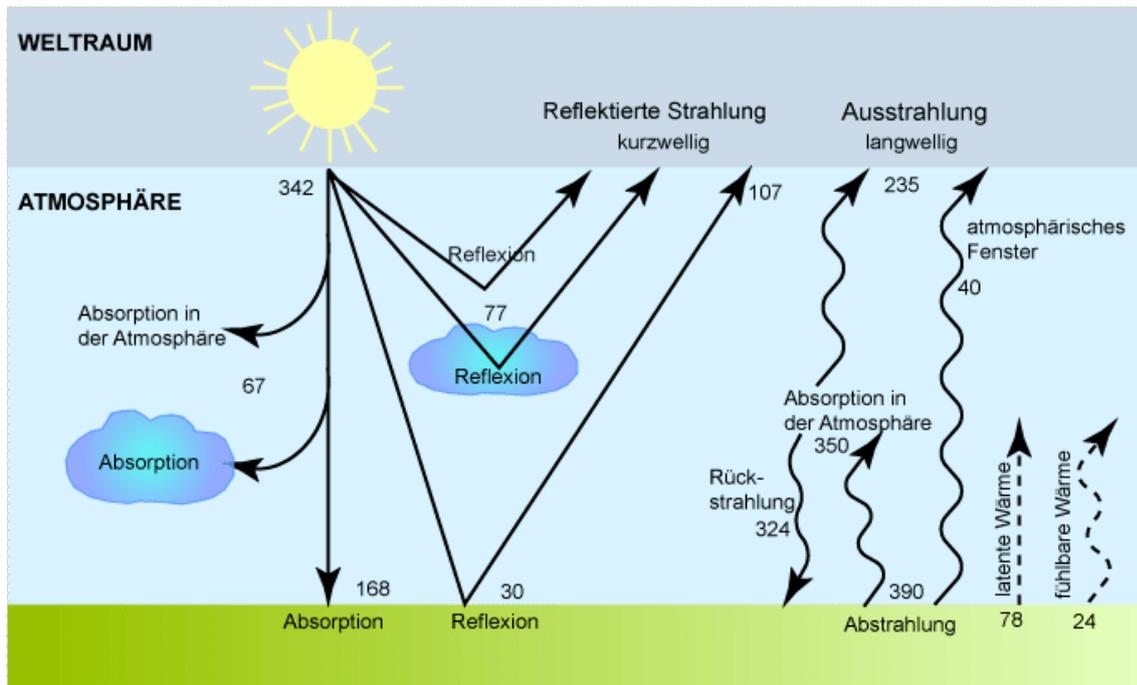


Abbildung 2.5: Skizze der Flüsse der Strahlungsbilanz der Erde. Alle angegebenen Werte sind in Wm^{-2} .

3 Methodik

3.1 Aufbau des verwendeten Generative Adversarial Network

Das GAN besteht aus einem U-Net als Generator Netzwerk und einem Multi-Scale Diskriminator (Schmidt et al., 2020). Die Architektur des Generators ist motiviert durch die Veröffentlichung von Ronneberger et al. (2015). Der Name der Architektur ergibt sich aus der Form, wie in Abbildung 3.1 zu erkennen. Diese besteht links aus einem kontrahierenden Pfad und rechts aus einem expandierendem Pfad. Die linke Seite hat die typische Architektur eines Convolutional Network, wie in Abbildung 2.1 dargestellt. In diesem werden Information über das Bild gesammelt, jedoch auf Kosten der räumlichen Information, wo typische Merkmale im Bild vorhanden sind. In diesem Fall wird für die Faltung eine 3×3 Kernel Matrix verwendet auf welche jedes mal die Aktivierungsfunktion ReLU folgt. Dies resultiert im ersten Layer zu 64 Feature Maps. Bei jedem Downsampling Schritt wird ein 2×2 Max Pooling angewendet und die Anzahl der Feature Maps verdoppelt. In dem expandierendem Pfad rechts besteht jeder Schritt aus einem upsampling der Feature Map gefolgt von einer 2×2 Faltung, dass die Anzahl der Feature Maps halbiert. Die entstehenden Matrizen werden mit den dazugehörigen Matrizen aus dem linken Pfad verbunden. So können die Informationen aus dem kontrahierenden Pfad lokalisiert werden. Der Output ist dann eine dreidimensionale Matrix mit der Größe des Bildes, sowie der Anzahl von Klassifizierungen. In dem Fall von Ronneberger et al. (2015) sind dies zwei.

Die Architektur des verwendeten Diskriminators ist ein Multi-Skalen Netzwerk beschrieben von Wang et al. (2018). Genauer sind es drei Diskriminator Netzwerke, welche alle gleich aufgebaut sind. Der Unterschied ist die Skala des Inputs, auf denen die Netzwerke arbeiten. Eins der Netzwerke bewertet den Input auf der Skala des Originalbildes, während bei den anderen die Skala des Originalbildes mittels 2×2 MeanPooling um zwei bzw. vier runter skaliert wurde. Der Hintergrund warum mehrere Skalen im Diskriminator verwendet werden liegt daran, dass dem Generator sowohl im globalen Kontext, als auch im Detail, eine bessere Anleitung geben wird.

3.2 Training

Der Generator wird trainiert, indem eine gewichtete Kostenfunktion mit zwei Verlusten angewendet wird. Diese sind ein non-saturating GAN loss und ein matching loss. Als non-saturating GAN loss wurde mit dem hinge loss und dem least square loss (LSGAN) experimentiert. Die besserer Performance wurde mit dem LSGAN erzielt (Schmidt et al., 2020), weshalb der hinge loss nicht näher betrachtet wird.

In regulären GANs wird der Diskriminator mit der Sigmoid Cross Entropy Funktion trainiert. Dies kann jedoch im Training zu dem Problem führen, dass der Diskriminator zu

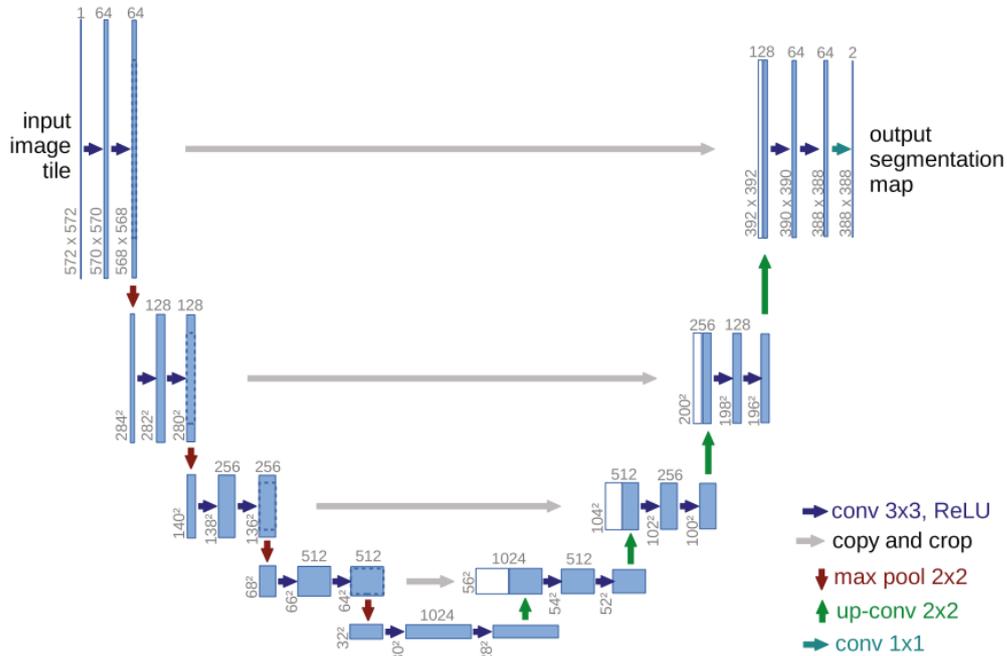


Abbildung 3.1: Darstellung der U-Net Architektur des Generators aus Ronneberger et al. (2015).

stark wird und der Generator keine Bilder mehr produzieren kann, auf die der Diskriminator reinfällt. In diesem Fall kann der Generator sich nicht mehr verbessern. Aus diesem Grund wird der least square loss verwendet. Dieser wirkt dem ganzen entgegen, indem die Pearson χ^2 Divergenz minimiert wird. Weiterhin gibt es mit LSGAN zwei Vorteile. Die erzeugten Bilder haben eine höhere Qualität und das Netzwerk bleibt während des Trainings stabiler (Mao et al., 2017).

Als matching loss wird der L^1 matching loss zwischen dem generiertem und dem Bild aus dem Datensatz verwendet. Dies führt dazu, dass der Generator Bilder produziert, die nah an dem beobachteten Daten sind (Schmidt et al., 2020). Der L^1 Verlust wird dem L^2 Verlust vorgezogen, da bei der Methode mit dem L^1 Verlust die erzeugten Bilder weniger verschwommen sind (Isola et al., 2018).

Der verwendete Trainingsdatensatz besteht aus 3100 Wertepaaren $\chi = \{m_i, r_i\}$ aus meteorologischen Daten m_i und den dazugehörigen Reflexionsdaten r_i . Die meteorologischen Daten m_i sind dreidimensionale Matrizen der Größe $44 \times 256 \times 256$, bestehend aus 42 Messungen aus MERRA-2 wie z.B. Wind, Temperatur oder Luftfeuchtigkeit. Da es sich um Satellitendaten handelt und der Satellit polarumlaufend ist, werden Daten über die Position in geographischer Länge und Breite des Satelliten relativ zu Erde benötigt. Die Reflexionsdaten r_i sind ebenfalls dreidimensionale Matrizen mit den Dimensionen $3 \times 256 \times 256$, welche die Reflexion gemessen bei den RGB Wellenlängen (680, 550 und 450 nm) abgespeichert hat (Schmidt et al., 2020).

4 Ergebnisse

4.1 Visuelle Analyse

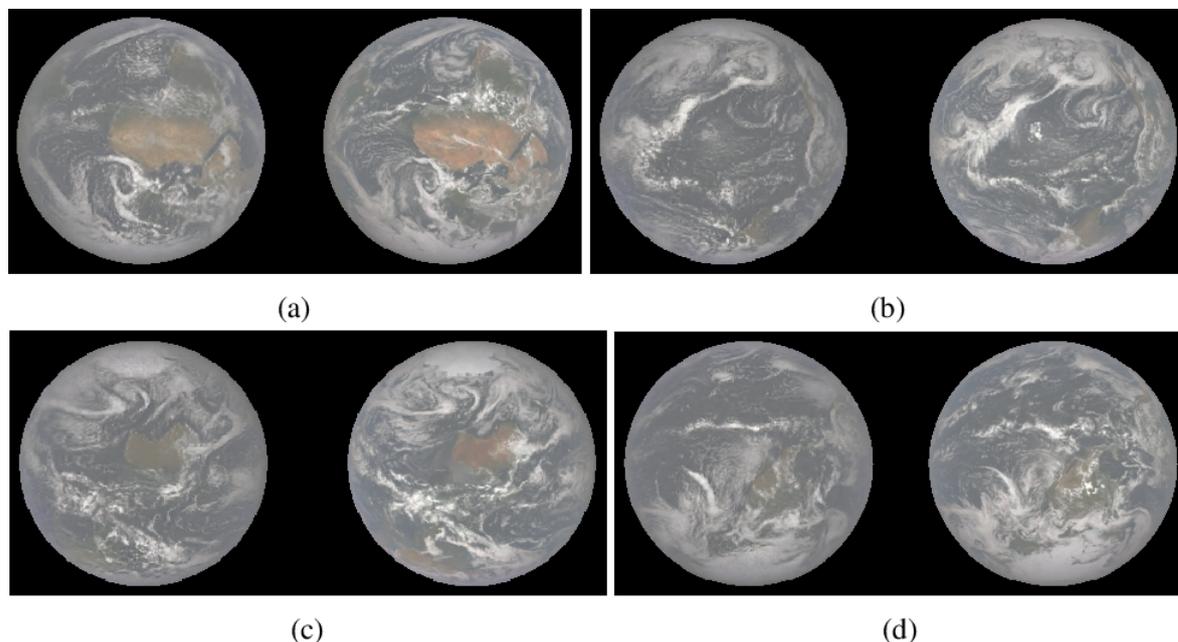


Abbildung 4.1: Vier verschiedene vom GAN generierte Bilder (links) und deren dazugehörigen beobachteten echten Bilder (rechts).

Die in Abbildung 4.1 dargestellten Bildpaare von generierten Bildern und den dazugehörigen echten Bildern $(G(m_i), r_i)$ zeigen, dass das Modell sowohl grobe als auch feine Strukturen der Wolkenformationen generieren kann. Ebenfalls sind die Kontinente und Ozeane in den wolkenfreien Bereichen gut wiederzuerkennen. Wird das Paar (b) näher betrachtet, so ist zu erkennen, dass der grobe Struktur der Wolken des Wirbels oben rechts, sowie das Wolkenband, das damit verbunden ist, gut abgebildet sind im Modell. Bei genauerer Untersuchung des Bildpaares ist zu erkennen, dass die feinen Strukturen im Modell nicht mit denen des echten Bildes übereinstimmen. Ähnliches ist auf allen dargestellten Bildpaaren zu erkennen. Zur genaueren Untersuchung des Modells wurden drei Bilder aus denselben Messungen generiert und verglichen. Auch in diesem Fall ist die globale, und somit grobe, Struktur konsistent gut bei allen drei generierten Bildern mit leichten Unterschieden in den feinen Strukturen.

Ebenfalls kann das Training des Modells quantitativ bewertet werden. Dies wird gemacht, um die Diversität des Modells bewerten zu können. Dazu wurden fünf Stichproben manuell ausgewählt, um mehrere Orte der Erde abzubilden und jeweils drei Bilder generiert. Mit den drei Metriken des Pixelweise Mittelwerts, der Standardabweichung und des Interquartilsabstand ergibt sich die Abbildung 4.2.

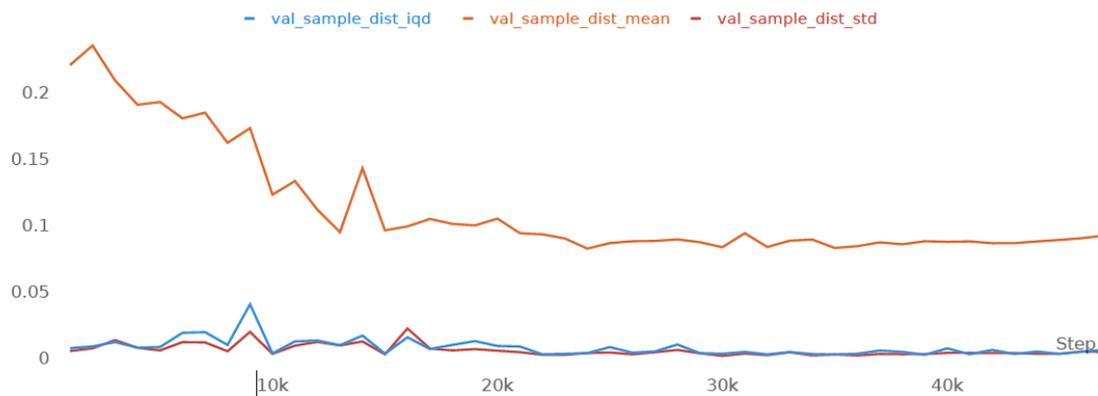


Abbildung 4.2: Entwicklung der drei verschiedenen Metriken Interquartilsabstand (blau), Pixelweise Mittelwert (orange) und der Standardabweichung (rot) über die Anzahl der Generationen.

Es ist zu erkennen, dass der Interquartilsabstand, sowie die Standardabweichung über die Generationen von Bildern sich nur wenig ändert und konstant niedrig ist. Anders ist es mit dem Pixelweise berechneten Mittelwert. Der Wert sinkt über die ersten 15.000 bis 20.000 Generationen, ehe der Mittelwert nahezu konstant bleibt. Je mehr Generationen es gibt, desto besser ist das generierte Bild, was jedoch auf Kosten der Diversität geht.

4.2 Spektrale Analyse

Neben der Visuellen Analyse zwischen den generierten Bildern und den Beobachtungsdaten können die Frequenzspektren zwischen Modell und Realität verglichen werden. Die Spektren berechnen sich durch die 2D Diskrete Fourier Transformation (DFT). So können geometrische Übereinstimmungen auf den Bildern objektiver durch die berechneten Spektren bestimmt werden. Dazu werden die Amplituden aus der 2D DFT der grauskalierten Bilder (Modell und Realität) bestimmt und die Histogramme dieser Amplituden ausgewertet. Dies ist in Abbildung 4.3 zu erkennen. Dargestellt sind die Amplituden und deren Histogramme für bestimmte Bildausschnitte von Realität und Modell. Der Visuelle Vergleich zeigt bereits, dass das generierte Bild des Modells gut ist. Anhand der 2D DFT und der daraus resultierenden Histogrammen kann dies ebenfalls quantitativ bestimmt werden, denn die Histogramme sind nahezu identisch. Es fällt hauptsächlich auf, dass die Mittelwerte beider Verteilungen unterschiedlich sind. Im Modell ist dieser minimal kleiner als in den Beobachtungsdaten.

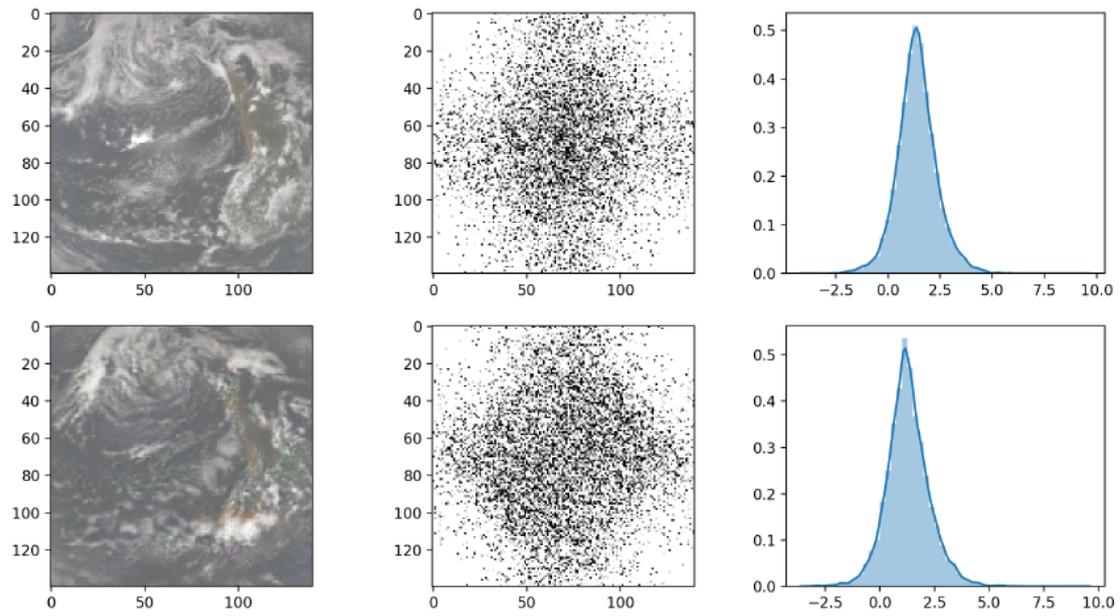


Abbildung 4.3: Spektrale Analyse von Beobachtungsdaten (oberer Reihe) und Modelldaten (untere Reihe). Im einzelnen sind Ausschnitte des Bildes (links) mit den dazugehörigen Amplituden der 2D DFT (mitte) und deren Histogrammen (rechts)

5 Zusammenfassung und Ausblick

Wie in Kapitel 4 zu erkennen, sind die generierten Bildern aus dem verwendeten Generative Adversarial Network Vergleichbar mit den dazugehörigen Beobachtungsdaten. Um die damit verbundene Forschungsfrage aufzugreifen, ob GANs genutzt werden können, um Cloud Reflective Fields zu modellieren, so kann dies bedingt mit Ja beantwortet werden. Es ist definitiv ein effektiver Ansatz zum Aufbau eines datenbasierten Rahmen. Schmidt et al. (2020) sind der Meinung das solch genierten Bilder dazu genutzt werden können Berechnungszeiten bei der Wolkenmodellierung in Klimamodellen zu verringern. Jedoch muss beachtet werden, dass das Modell Grenzen hat, wie die Unschärfe oder kleinskalige Schachbrett Artefakte. Dies könnte verbessert werden, indem der Trainingsdatensatz vergrößert wird und die Trainingszeit verlängert wird. Ebenfalls könnten Verbesserungen durch Hyperparameter Tuning, inklusive Änderungen der Architektur von Generator oder Diskriminator, erzielt werden. Weiterhin sind die aktuellen generierten Bilder einzige Aufnahme ohne zeitliche Strukturen. Dies soll in Zukunft durch das Hinzufügen von Datum und Zeit als zusätzliche Labels verändert werden. Ein weiterer Punkt war, dass die Vielfältigkeit gering war. Ein Ansatz dies zu lösen, ist das Hinzufügen von Rauschen beim Input. So wird kann die Diversität erhöht werden.

Alles in allem ist dies ein valider Ansatz Generative Adversarial Networks im Zusammenhang mit der Klimaforschung einzusetzen, da so die Modelle durch besser aufgelöste Wolken, bessere Ergebnisse liefern können ohne eine zu Hohe Rechenkapazität zu benötigen.

Literaturverzeichnis

Convolutional Neural Network (2021). Convolutional neural network — Wikipedia, the free encyclopedia. [Online; Zugriff 12.08.2021].

Ebert, D. (2019). Bildklassifizierung mit neuronalen netzen.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial networks.

Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2018). Image-to-image translation with conditional adversarial networks.

Kim, P. (2017). *MATLAB Deep Learning*. Apress.

Mao, X., Li, Q., Xie, H., Lau, R. Y., Wang, Z., and Paul Smolley, S. (2017). Least squares generative adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Neuronale Netze (2021). Neuronale Netze - Eine Einführung – Aktivitätsfunktion, Aktivitätslevel und Output. [Online; Zugriff 12.08.2021].

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Lecture Notes in Computer Science*, pages 234–241. Springer International Publishing.

Schmidt, V., Alghali, M., Sankaran, K., Yuan, T., and Bengio, Y. (2020). Modeling cloud reflectance fields using conditional generative adversarial networks.

Wang, T.-C., Liu, M.-Y., Zhu, J.-Y., Tao, A., Kautz, J., and Catanzaro, B. (2018). High-resolution image synthesis and semantic manipulation with conditional gans.
