Inhaltsverzeichnis

1 Einleitung

Hier eine kurze Einleitung in welchem Rahmen diese Arbeit entstanden ist und schonmal ganz kurz auf SETI eingehen.

2 SETI Breakthrough Listen

Die kaggle Challenge *SETI Breakthrough Listen - E.T. Signal Search* war ein öffentlicher Machine Learning Wettbewerb des *Berkeley SETI Research Centers* im Zeitraum vom 10. Mai 2021 bis 18. August 2021. Die zugrunde liegenden Daten sind noch verfügbar, sodass Interessierte sich nach wie vor mit diesem Problem beschäftigen können. Im folgenden werden wir die Challenge stets abgekürzt als *SETI* bezeichnen.

Die Herausforderung bei *SETI* besteht darin, Spektrogramme, also eine bildliche Darstellung eines Frequenzbereichs in einem bestimmten Zeitraum, die basierend auf Rohdaten des *Green Bank Telescopes* generiert worden sind, auf das Vorkommen von künstlich hinzugefügten extraterrestrischen Signalen zu untersuchen. Hierbei ist es wichtig, diese Signale von irdisches Signalen, wie etwa einem Radiosignal, zu unterscheiden. Um diese Unterscheidung vornehmen zu können, sind jeweils sechs Spektrogramme zusammengefasst, wobei die Spektrogramme eins, drei und fünf jeweils Aufnahmen des zu untersuchenden Ziels „A“ sind und die übrigen jeweils auf Aufnahmen eines anderen Himmelskörpers „B“, „C“ und „D“. Eine solche Gruppe von Spektrogrammen (ABACAD) wird bei *SETI* als *Kadenz-Ausschnitt*, im folgenden nur noch „Kadenz“, bezeichnet. Jedes Spektrogramm zeigt den Frequenzbereich für einen Zeitraum von fünf Minuten, eine Kadenz stellt folglich einen Beobachtungszeitraum von 30 Minuten dar.

Abbildung 1 zeigt ein Beispiel für eine Kadenz mit einem extraterrestrischen Signal. Die drei Spektrogramme in der oberen Zeile sind *on target*, auf diesen ist im Frequenzbereich, welcher durch die x-Achse repräsentiert ist, zwischen 150 und 200 ein Signal zu erkennen, welches auf den Spektrogrammen in der unteren Zeile, welche *off target* sind, nicht zu sehen ist. Die grüne senkrechte Linie, die auf allen sechs Spektogrammen zu sehen ist, ist hingegen ein irdisches Signal. Offensichtlich muss ein Signal nicht auf jedem der drei *on target* Spektrogrammen zu sehen sein, da ein Signal nicht zwingend über den gesamten zeitlichen Betrachtungsraum aktiv sein muss.

Die Trainingsdaten für *SETI* enthalten 60.000 Kadenzen (*Heuhaufen*) von denen 6.000 *Nadeln* sind, also Kadenzen, die ein künstlich eingefügtes extraterrestrisches Signal enthalten. Einige dieser Signale sind bei entsprechender Visualisierung sofort mit bloßem Auge zu erkennen, andere sind in, durch irdische Signale verursachten, Rauschen versteckt.

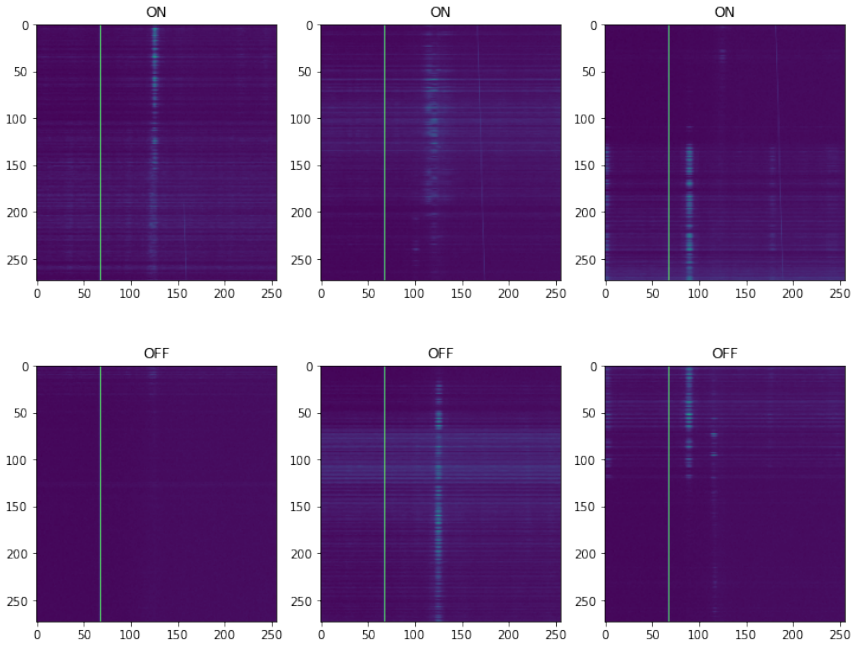

Abbildung 1: Beispiel für ein extraterrestrisches Signal

3 Implementierung

Im folgenden beschäftigen wir uns nun mit der Problemlösung für *SETI*. Wir schauen uns erste Ansätze mit reiner Computer Vision an, die uns helfen sollen auch versteckte Signale extrahieren zu können, um sie mit bloßem Auge erkennen zu können. Wir haben uns für diesen Einstieg entschieden, um ein Gefühl für die visuelle Form der gesuchten Signale und für den Datensatz allgemein zu erhalten. Im darauf folgenden Kapitel werden wir uns mit *Convulutional Neural Networks*, kurz *CNNs*, beschäftigen.

3.1 Erste Ansätze mit Computer Vision

Hier können wir unseren ersten Ansätze mit reiner Computer Vision beschreiben und zunächst darauf eingehen, wie sich traditionelle Computer Vision von Machine Learning unterscheidet / abgrenzt und warum wir zunächst mit CV experimentiert haben.

3.1.1 Implementierte CV Filter

Hier können wir die Filter, die wir am Anfang implementiert haben jeweils kurz beschreiben

3.1.2 Zwischenfazit: CV alleine bringt keinen Erfolg

Kurzes Zwischenfazit, das CV alleine nichts nützt, weil die Signale selbst nach der Anwendung unserer Filter auf vielen Positives nicht zu extrahieren sind.

4 Deep Learning

Natürlich wollen wir nicht alle Kadenzen einzeln manuell betrachten und entscheiden, ob sie eine Nadel enthalten oder nicht. Vielmehr wollen wir ein Machine Learning Model trainieren, das uns diese Arbeit abnimmt und Nadeln findet, die wir gar nicht entdecken würden. Hierzu wollen wir ein *Convolutional Neural Network (CNN)* trainieren, das die Kadenzen in genau zwei Klassen einteilt: enthält eine Nadel oder enthält keine Nadel.

4.1 Convolutional Neural Networks

Hier können wir uns nochmal überlegen oder mit Herrn Baier besprechen, wie doll wir bei den Erklärungen zu den einzelnen Punkten von CNNs ins Detail gehen sollen. Vll können wir auch vieles als bekannt voraussetzen und uns mehr auf unsere Implementierung konzentrieren. Dies wäre insgesamt vll nochmal abzuklären auch für die CV Section.

- Gewichte
- Loss Function
- Gradient
- Optimizer
- Traning und Validierung
- Splitten der Trainingsdaten
- Dataloaders
- Metriken (Accuracy, Precision, Recall, F1 Score, Roc Auc Score)

4.2 Transfer Learning

4.2.1 efficientnet

4.3 Imbalance

4.4 Scheduler

4.5 Folds

5 Tech Stack

Zusammenfassung der verwendeten Tools und Bibliotheken, die teilweise auch vorher im Text schon genannt wurden.

6 Fazit