

Exposé - Bachelorabschlussarbeit: Untersuchungen von Style Transfer Methoden für die Verwendung mit leistungsarmer Hardware

Christoph Stach

07.01.2019

Abstract

Die Abschlussarbeit im Bereich der Angewandte Informatik handelt darum verschiedene Style Transfer Methoden zu implementieren, die Konzepte zu verstehen und Sie auf Tauglichkeit für Systeme mit leistungsarmer Hardware zu testen. Style Transfer ist ein Bereich des Deeplearnings. Es werden dabei zwei, oder mehr, willkürliche Bilder zu einem neuen Bild kombiniert. Beim ersten Bild wird der Inhalt aus dem Bild herausgefiltert, z.B. aus einem Foto auf dem ein Vogel zu sehen ist. Aus dem zweiten Bild wird Stil herausgefiltert, z.B. aus einem abstrakt gezeichneten Gemälden von Vincent van Gogh. Als Ergebnis entsteht ein neues Bild, welches den gleichen Vogel darstellt, als wäre er von Vincent van Gogh gemalt worden. Es sollen unterschiedliche Algorithmen untersucht werden und die Ergebnisse mit einander verglichen werden.

1 Motivation

Ich bin ausgebildeter Webentwickler und spezialisiere mich auf Machinelearning an der HTW Berlin. Bei meiner täglichen Arbeit, entwickle ich Anwendungen die performant im Browser ausgeführt werden. Das weckt das Interesse bei mir beide Gebiete zu kombinieren und die Ausführung von Machinelearning Modellen auf leistungsarmer Hardware zu untersuchen.

2 Zielsetzung

Ziel des Projektes ist ein Softwaresystem zu erstellen das in der Lage ist Style Transfer auf Geräten mit leistungsarmer Hardware (z.B. Mobiltelefone oder Laptops ohne GPU) zu ermöglichen.

3 Vorgehen

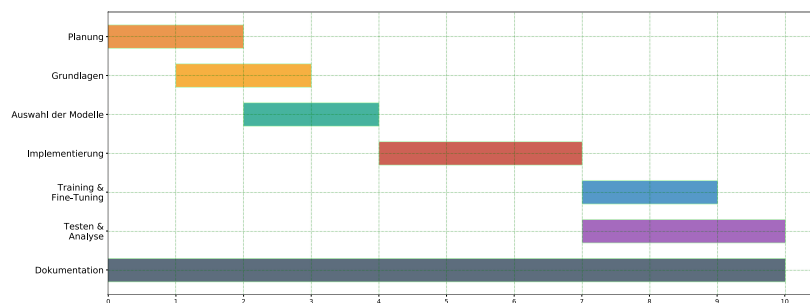
Nach der Einarbeitung in die Grundlagen von Style Transfer [**DBLP:journals/corr/GatysEB15a**] Methoden, müssen entsprechende Algorithmen gefunden werden. Da es sich um die Manipulation von Bildern handelt, werden vermutlich Convolutional Neural Networks [**lecun-gradientbased-learning-applied-1998**], welche sich dazu eigenen Features aus Bildern zu extrahieren, zum Einsatz kommen. Im nächsten Schritt muss ein Datensatz in entsprechender Größe generiert oder gefunden werden, mit dem ein solches Netzwerk trainiert werden kann. Auch denkbar ist der Einsatz von Transfer Learning mit bereits bestehenden, vortrainierten Netzwerkarchitekturen [**DBLP:journals/corr/SimonyanZ14a**], um die Ergebnisse zu verbessern. Letztendlich muss das Netzwerk auf den verwendeten Datensatz durch Backpropagation [**doi:10.1162/neco.1989.1.4.541**] optimiert werden.

Die Implementierung des Models in den Prototypen eines Softwaresystems soll lediglich der Veranschaulichen der Ergebnisse dienen.

4 Erfordernisse und Randbedingungen

Da das Berechnen von Deep Neural Networks und Convolutional Neural Networks sehr rechenintensiv sein kann, werden diese normalerweise auf Serversystemen mit eingebauter Graphic Processing Unit trainiert und ausgeführt. Das soll bei der Evaluation der Modelle berücksichtigt werden, damit Ergebnisse in einem vertretbaren Zeitaufwand auf Systemen ohne GPU zu berechnen sind.

5 Zeitplan



Das Gantt-Diagramm stellt den vorläufigen Zeitplan des Projektes da. Dieser kann sich jedoch während des Projektverlaufes noch ändern und wird bei Bedarf angepasst. Besonders zu beachten ist, dass das Training eines Neural Networks in Verbindung mit großen Datenmengen viel Zeit in Anspruch nehmen kann. Deswegen wurden für das Training vorläufig zwei Wochen veranschlagt.

6 Erwartete Ergebnisse

Das Ergebnis des Projektes ist der lauffähige Prototyp eines Softwaresystems mit der dazugehörigen Dokumentation und Erklärung aller verwendeten Algorithmen und Modelle.

7 Vorläufige Gliederung

1. Zusammenfassung (Abstract)
2. Einleitung
3. Grundlagen
4. Analyse
5. Konzeption
 - 5.1. Suche und Auswahl geeigneter Modelle
 - 5.2. Implementierung einer Basisversion der Modelle
 - 5.3. Verbesserungen und Erweiterungen an Modellen
 - 5.4. Hyperparameter-Tuning und Training
 - 5.5. Implementierung der Modelle in den Prototypen eines Softwaresystems
6. Implementierung
7. Test
8. Evaluation
9. Fazit / Ausblick