



Hochschule für Technik und Wirtschaft Dresden

Fakultät Informatik/Mathematik

newwebtechnology

Projektbericht

Thema:

**Erstellen eines anpassbaren Systems zur Erzeugung
von neuronalen Netzen für die Erkennung
vordefinierter Merkmale in Satellitenbildern**

Vorgelegt von: Sebastian Mischke
Dorfstraße 8, 01257 Dresden
geb. am 09.11.1995 in Dresden
Matrikelnummer: 37612

Studiengang: Medieninformatik

Betreuender Prüfer: Prof. Dr. Marco Block-Berlitz
HTW Dresden

Externer Betreuer: Ann-Christin Storms
New Web Technology GmbH

Inhaltsverzeichnis

1	**Einleitung	2
2	**Problemstellung	2
3	Projektumsetzung	2
3.1	**Adressdatei analysieren und laden als CSV	2
3.2	Visualisierung der Daten	2
3.3	Deeplearning Framework	2
3.4	Einarbeitung in Keras	2
3.5	Satellitenbilder laden	3
3.6	Neurales Netz	3
3.7	Trainingsprozess	3
3.8	Speichern und Laden von trainierten Netzen	3
3.9	Anpassen der Bilddaten	4
3.10	Erzeugen von Trainingsvorgaben	4
3.11	GPS-Koordinaten oder Adressen als Positionseingabe	4
3.12	Zusammenfügen von Bildausschnitten zu Gesamtbild	5
3.13	Speichern und Laden von Trainingsdaten in SQL	5
4	**Fazit	5

1 ****Einleitung**

- Zeitraum
- Unternehmen, Standort
- Abteilung
- Tätigkeitsbereich
- Sehr knapp aufgaben aufzählen
- Warum dieses Unternehmen?
- Wie bist du auf dieses Unternehmen gekommen?

2 ****Problemstellung**

- Adressliste von Häusern, welche zu klassifizieren sind
- Leichte Anpassung der Adressliste
- Leichte Anpassung der gewünschten zu erkennenden Merkmale
- Leichte Anpassung der Netzstruktur
- Gute Ergebnisvisualisierung

3 **Projektumsetzung**

3.1 ****Adressdatei analysieren und laden als CSV**

- Austauschformat der Adressliste ist CSV
- Einträge
 - Haus-ID
 - Straße
 - Hausnummer
 - Postleitzahl

- Stadt
- X-Koordinate
- Y-Koordinate

3.2 **Visualisierung der Daten**

Um einen besseren Überblick über die vorliegende Adressliste zu bekommen, sollte diese ähnlich einer 2D-Karte visualisiert werden. Hierzu wurden für jeden Eintrag die geographischen Koordinaten aus der Adressliste ausgelesen und separat in einer zweiten Liste gespeichert. Diese zweite Liste wird anschließend genutzt, um mit der Python-Visualisierungs-Bibliothek "matplotlib" ein Streudiagramm zu erstellen.

3.3 **Deeplearning Framework**

Da der Aufwand zur Erstellung eines eigenen Deeplearning-Frameworks zu hoch ist, soll ein bereits existierendes Framework verwendet werden. Deshalb wurde zuerst eine Liste von möglichen Frameworks sowie deren Vor- und Nachteile zusammengestellt. Nach anschließendem Abgleich mit den gegebenen Anforderungen, fiel die Wahl auf das Framework „Keras“.

3.4 **Einarbeitung in Keras**

Um Erfahrung mit der Verwendung von „Keras“ zu sammeln, wurde für die Einarbeitung zunächst mit der Erstellung eines einfachen neuronalen Netzes begonnen. Dieses sollte anhand der geographischen Koordinaten die Postleitzahl eines Hauses erkennen. Die dafür benötigten Informationen gingen aus der Adressliste hervor.

Nach dem Training des Netzes wurde der komplette Eingabedatensatz vom Netz

klassifiziert und die Ergebnisse zusammen mit den Eingabedaten in einer neuen Liste gespeichert. Die Visualisierung dieser Liste geschah anschließend wieder als Streudiagramm. Die Klassifizierungsergebnisse wurden dabei jeweils verschiedenen Farbwert zugeordnet.

3.5 Satellitenbilder laden

Es ist nicht immer möglich, auf einen vollständigen Bilddatensatz von Häusern zugreifen zu können. Deshalb sollte eine eigene Lösung entwickelt werden, um das benötigte Bildmaterial zu erzeugen. Hierfür wurden verschiedene Online-Kartendienste miteinander verglichen, die Entscheidung fiel auf Google Maps. Nach der Einarbeitung in die API von Google Maps, erfolgte die Implementierung einer Funktion, mit deren Hilfe die benötigten Informationen korrekt eingesetzt werden können. Die Funktion für jeden Eintrag der Adressliste ausgeführt um die geladenen Bilder für die spätere Verwendung lokal zu speichern.

3.6 Neuronales Netz

Neben dem Satellitenbild können zu Häusern noch zusätzliche Meta-Daten existieren. Diese bieten die Möglichkeit der Verbesserung der Ergebnisgenauigkeit des Netzes. Deshalb muss die Netzstruktur besonders an das Eingabeformat der Daten angepasst werden. Dazu wird das Netz aus mehreren Teilnetzen zusammengesetzt. Dies ermöglicht mehrere Eingabeschichten und damit die Nutzung verschiedener Eingabeformate. Das erste Teilnetz soll die Bilddaten analysieren, weshalb es die typischen Struktur eines Convolutional Neural Network, bestehend aus Faltungsschichten am Anfang und vollver-

netzte Schichten am Ende, erhält. Das zweite Teilnetz soll die Meta-Daten analysieren und besteht somit ausschließlich aus vollvernetzte Schichten. Das abschließende Teilnetz soll die beiden Eingabe-Netze verbinden und basierend auf deren Ergebnissen die Ausgabeschicht berechnen.

3.7 Trainingsprozess

Um den Ablauf des Trainingsprozesses des Netzes besser analysieren zu können, ist eine Visualisierung der Trainingsstatistiken, wie zum Beispiel die Klassifizierungsgenauigkeit, notwendig. Dafür werden die benötigten Informationen bereits während des Trainingsprozesses gespeichert. Im Anschluss daran werden alle auf diese Weise erfassten Daten in eine eigens dafür angefertigte Funktion gegeben, welche daraus abermals mit Hilfe von matplotlib ein Kurvendiagramm über die Trainingsepochen erstellt.

3.8 Speichern und Laden von trainierten Netzen

Der Trainingsvorgang eines neuronalen Netzes ist sehr zeitaufwändig. Um also trainierte Netze zu verwenden oder einen Trainingsprozess unterbrechen und fortsetzen zu können, muss das Speichern und Laden eines Netzes möglich sein. Dafür werden die Struktur des Netzes und die Gewichte aller Schichten in zwei separaten Dateien gespeichert. Dies ermöglicht es, ein vollständiges trainiertes Netz zu laden, aber auch, eine Netzstruktur für einen anderen Trainingsprozess wiederzuverwenden.

3.9 Anpassen der Bilddaten

Die vorliegenden Bilddaten entsprechen nicht immer den Anforderungen des Trainingsprozesses. Deshalb soll es möglich sein, die Bilder vor Beginn des Trainingsprozesses zu bearbeiten. Dazu wurde eine Funktion erstellt, welche ein lokal vorliegendes Bild lädt und nach besonderen Parametern editiert. So kann ein Bild skaliert, beschnitten oder in seinem Seitenverhältnis geändert werden. Das Speichern des Ergebnisbildes zur Wiederverwendung ist möglich, aber sehr speicheraufwändig, da aus jeder Parameterkombination ein neues Bild hervorgeht. Darum wird das Ergebnisbild direkt in den Trainingsprozess gegeben, wodurch ein langfristiges Speichern nicht notwendig ist.

3.10 Erzeugen von Trainingsvorgaben

Innerhalb der Trainingsdaten müssen die gewünschten Merkmale für jedes Haus als Trainingsvorgabe vorliegen. Sollte es sich dabei um Merkmale handeln, zu denen noch keine Trainingsvorgaben vorliegen, müssen diese händisch erzeugt werden. Dafür wurde eine eigenständige Anwendung entwickelt, welche dem Nutzer das Erzeugen der Trainingsvorgaben erleichtert. Um bereits implementierte Funktionen wiederverwenden zu können, wurde die Anwendung in Python mit dem GUI-Toolkit „Tk“ umgesetzt.

In der Anwendung wird ein Satellitenbild, sowie eine Liste von möglichen Merkmalen angezeigt. Der Nutzer kann die Merkmale auswählen, die für das gesamte Bild beziehungsweise das gezeigte Haus im Bild zutreffen. Diese Zuordnung wird dann in einer CSV-Datei gespeichert. Ist

der Nutzer mit der Merkmalszuordnung für ein Bild fertig, wird der nächste Adresslisteneintrag gezeigt.

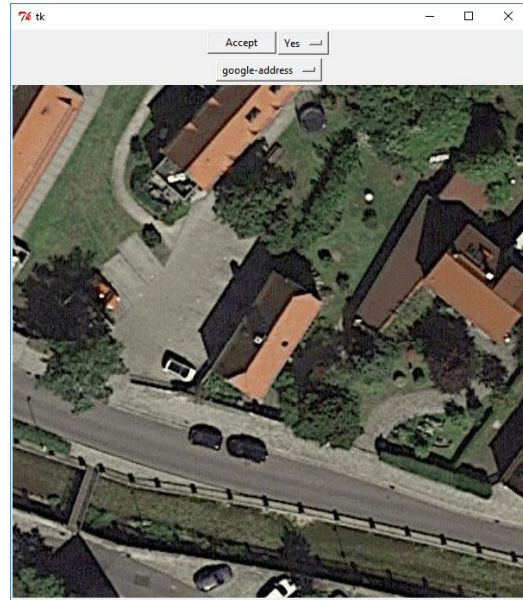


Abbildung 1: Interface

3.11 GPS-Koordinaten oder Adressen als Positionseingabe

Da es von Vorteil wäre, wenn die Liste von Häusern nicht zwingend deren geographischen Koordinaten enthalten muss, soll es nun noch zusätzlich die Möglichkeit geben, stattdessen die Adressen der Häuser zu benutzen. Dafür müssen die Adresstücke, welche in der Adressliste in Straße, Hausnummer, Postleitzahl und Stadt aufgeteilt sind, zusammengesetzt und in ein Format gebracht werden, welches von der Google Maps API genutzt werden kann. Dadurch können alle folgenden Schritte unabhängig von der Art der Positionsangabe wie bisher erfolgen.

Sollten sich in der Adressliste sowohl Koordinaten als auch die Adresse der Häu-

ser befinden, so kann der Funktion, welche die Bilder herunterlädt, die zu benutzende Positionsangabe als Parameter übergeben werden.

3.12 Zusammenfügen von Bildausschnitten zu Gesamtbild

Teilweise besteht der Bilddatensatz aus Bildern, die nur eine kleine Fläche abdecken. Dies könnte bei der Analyse von größeren Häusern oder Grundstücken ein Problem darstellen. Außerdem erhält das Netz in diesem Fall keine Informationen über die nähere Umgebung, wodurch Ungenauigkeiten hinsichtlich der Klassifizierungsquote zustande kommen können. Deshalb wurde eine Funktion implementiert, welche Satellitenbilder zusammensetzt, um damit ein Bild zu erzeugen, welches eine größere Oberfläche zeigt. Das Ursprungsbild, welches direkt die Hausadresse oder -koordinate beinhaltet, befindet sich dabei im Zentrum, damit das zu betrachtende Haus immer mittig im Ergebnisbild liegt.

3.13 Speichern und Laden von Trainingsdaten in SQL

Da sich das Speichern von Adresslisten inklusive der zugehörigen Meta-Daten als CSV-Datei langfristig betrachtet als unpraktisch erweisen könnte, sollen die benötigten Informationen aus einer SQL-Datenbank geladen werden. Dafür wurde eine Klasse erstellt, welche SQL-Tabellen und CSV-Dateien ineinander konvertieren kann. Somit bleibt die Option bestehen, CSV-Dateien weiterhin als Dateneingabe für das Netz zu verwenden.

Zudem können die Klassifizierungser-

gebnisse des Netzes, welche bisher ebenfalls in Form einer CSV-Datei erzeugt wurden, in der Datenbank gespeichert werden. Wird die Arbeit mit mehrerer Computern gleichzeitig durchgeführt, können so verschiedene Bilder parallel klassifiziert und in der Datenbank zusammengeführt werden.

4 **Fazit

- Was hat dir an deinem Praktikum besonders gut gefallen? Was hat dich an der Arbeit vielleicht sogar überrascht?
- Was könnte in Zukunft an diesem Praktikum (z. B. am Ablauf, an der Betreuung, an der Organisation) noch verbessert werden?
- Was hast du während des Praktikums neu gelernt und was wusstest du schon?
- War die Arbeit eher anspruchsvoll für dich oder hast du dich oft gelangweilt?
- Welche Stärken und Schwächen konntest du an dir selbst entdecken?
- Welche Auswirkungen hat das Praktikum auf deine spätere Berufs- oder Studienwahl?
- Welche Chancen siehst du für dich persönlich und im Allgemeinen in diesem Berufsfeld?
- Wo kannst du deine neu gewonnen Interessen in Zukunft eventuell noch weiter vertiefen?
- Würdest du das Praktikum weiterempfehlen und wenn ja, an was für eine Person mit welchen Eigenschaften?

- Inwieweit decken sich deine Erfahrungen mit dem, was du vor Beginn des Praktikums erwartet oder erhofft hast?