Hochschule für Technik und Wirtschaft Dresden

Fakultät Informatik/Mathematik

Bachelorarbeit

Thema:

Merkmalserkennung von Gebäuden und Grundstücken in Satellitenbildern mittels Deeplearning

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

Studiengang: Medieninformatik

Externer Betreuer: Ann-Christin Storms
New Web Technology GmbH

Betreuender Prüfer: Prof. Dr. Marco Block-Berlitz

Zweitgutachter: Prof. Dr. Hans-Joachim Böhme

Letzte Änderung: 17. Mai 2017

Abgabetermin:

Inhaltsverzeichnis

Verzeichnis verwendeter Abkürzungen

Nomenclature

API Eine Programmierschnittstelle, genauer Schnittstelle zur Anwendungsprogrammierung, häufig nur kurz API genannt (englisch application programming interface, wörtlich ‚Anwendungsprogrammierschnittstelle'), ist ein Programmteil, der von einem Softwaresystem anderen Programmen zur Anbindung an das System zur Verfügung gestellt wird.

MLP A multilayer perceptron (MLP) is a feedforward artificial neural network model that maps sets of input data onto a set of appropriate outputs. An MLP consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one. Except for the input nodes, each node is a neuron (or processing element) with a nonlinear activation function. MLP utilizes a supervised learning technique called backpropagation for training the network. MLP is a modification of the standard linear perceptron and can distinguish data that is not linearly separable.

Abbildungsverzeichnis

Zusammenfassung

Inhalt der Arbeit

1 Einleitung und Motivation

2 Konkretisierung der Aufgabenstellung

3 Stand der Technik

4 Gesamtplan

Das System zur Merkmalserkennung besteht aus drei Kernbausteinen. (siehe Abbildung 1)

- Den Satellitenbildern, die als Informationsgrundlage und Input-Daten für den Klassifizierer genutzt werden
- Dem künstlichen neuronalen Netz, welches die Input-Daten analysiert und klassifiziert
- Der Trainingsvorgabe, welche zum Trainieren des Netzes benutzt wird.

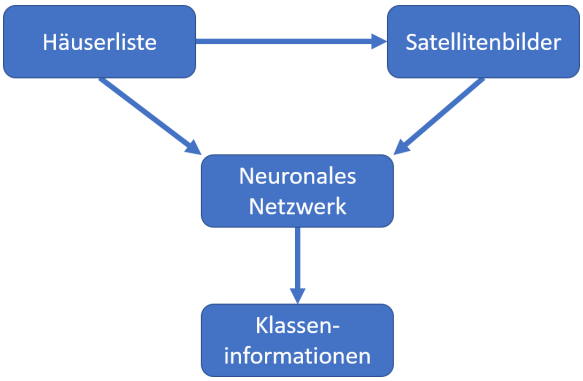

Abbildung 1: Datenflussdiagramm

4.1 Satellitenbilder

Die Satellitenbilder bilden die Grundlage der Input-Daten für das Netz. Da eine eigenständige Erzeugung einen viel zu hohen Aufwand bedeuten würde, werden stattdessen Bilder verwendet, die mittels der Google Static Maps API erzeugt wurden. Dafür sind entweder die GPS-Koordinaten oder die Adresse des Grundstückes notwendig.

4.2 Trainingsdaten

Trainingsdaten geben dem Netz den gewünschten Output vor und werden dementsprechend zu dessem Training verwendet.

Das Erzeugen der Trainingsdaten ist in der Regel ein aufwändiger Prozess und stark vom gewünschten zu erkennenden Merkmal abhängig.

4.3 Künstliches neuronales Netz

Das neuronale Netz besteht es drei Teilen. Einem CNN, welches die Satellitenbilder als Input-Daten bekommt und diese auswertet, einem MLP, welches zusätzliche Metadaten des Gebäudes bzw. Grundstückes als Input-Daten bekommt, und einem MLP, welches die Output-Daten des CNN und des MLP als Input-Daten bekommt.

Nach jeder Trainingsepoche werden die Validierungsdaten vom Netz klassifiziert und die Ergebnisse mit der Trainingsvorlage verglichen. Der sich daraus ergebende Genauigkeitswert, auch Accuracy genannt, wird dann mit dem bisherigen Bestwert verglichen. Sollte die neue Accuracy besser sein als der bisherige Spitzenwert, so werden die Gewichte des Netzes gespeichert und der neue Bestwert wird sich gemerkt.dadurch soll ein Overfitting des Netzes vermieden werden.

```
1 def main():
2     # Load csv
3     X_train, Y_train = load_csv("
          data.csv")
4     # Create net
5     model = create_net(X_train,
          Y_train)
6     # Train net
7     history = model.fit(x=X_train, y
          =Y_train)
8     # Save net
9     save_model(model, "structure.
          json", "weights.h5")
```

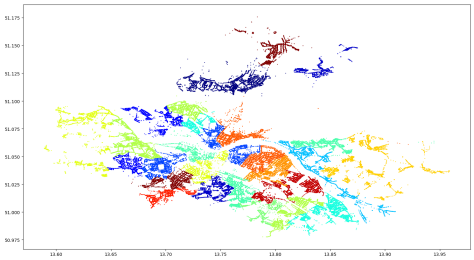

Abbildung 2: Point Plot Beispiel

5 Experimente

5.1 Postleitzahl

5.2 Solaranlagen

5.3 Schulen

6 Ergebnisse

7 Ausblick

Literatur