

Textextraktion und Einordnung mit Hilfe neuronaler Netze

Erstellung eines prototypischen Systems zur Extraktion und semantischen Einordnung von Texten aus Rastergrafiken im RGB Farbraum auf Basis neuronaler Netze am Beispiel einer Liste von Lebensmittelinhaltsstoffen

Gliederung

- Motivation - Ziel - Aufgaben
- Grundlagen
- Umsetzung
- Evaluierung
- Zusammenfassung
- Fragerunde und Vorführung



Motivation - Ziel - Aufgaben

Warum diese Anwendung?

- willkürliche Bilder bilden einen großer Teil des täglichen Datenaufkommens
- willkürliche Bilder enthalten Information in Form von Texten
- in Bildern eingebetteter Text entziehen sich einer einfachen Auswertung
- herkömmliche OCR-Verfahren kommen bei natürlichen Aufnahmen an ihre Grenzen
- neuronale Netze erweisen sich hier als außerordentlich erfolgreich



Motivation - Ziel - Aufgaben

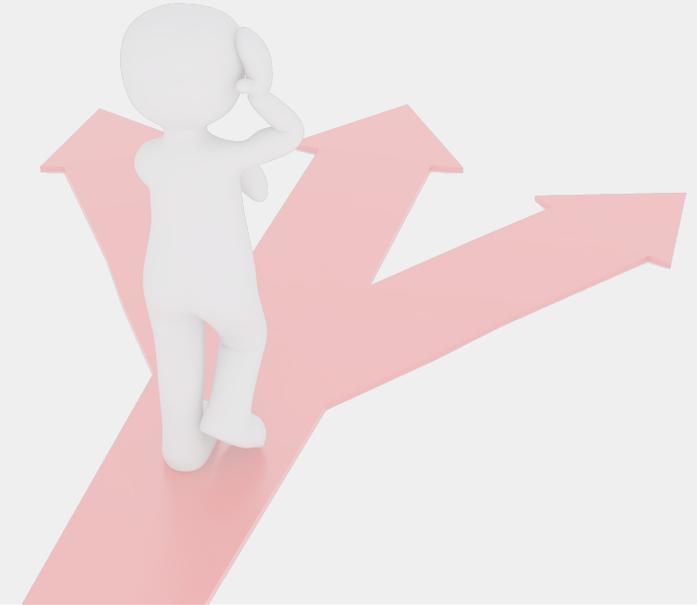
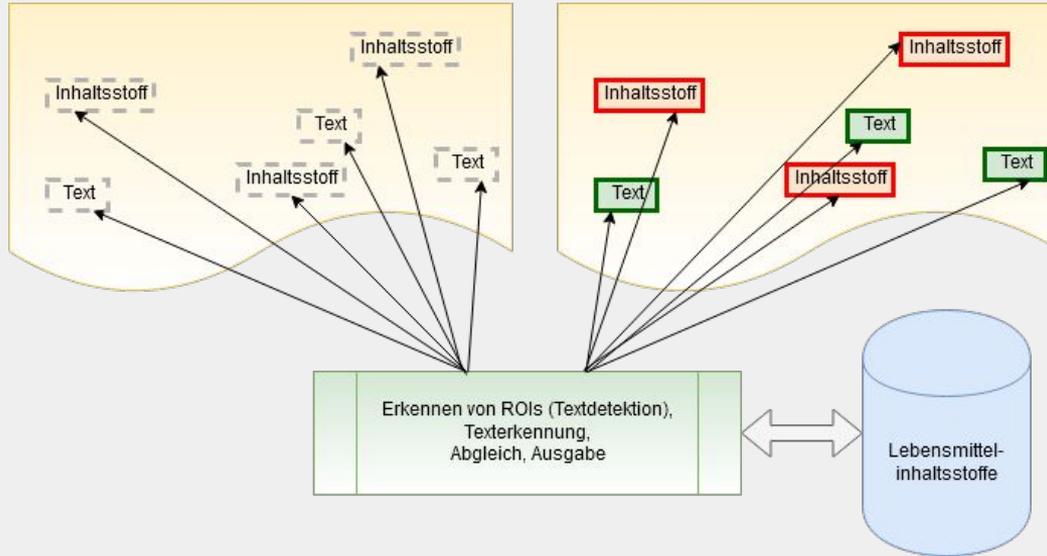
Was soll erreicht werden?

- erstellen einer prototypischen Komponente, die
 - Textelemente in Rastergrafiken detektiert und vorhergesagt
 - Texte anhand einer Datenbasis am Beispiel von Lebensmittelinhaltsstoffen abgleicht
 - erfolgreiche Abgleiche visuelle hervorhebt



Motivation - Ziel - Aufgaben

Was soll erreicht werden?

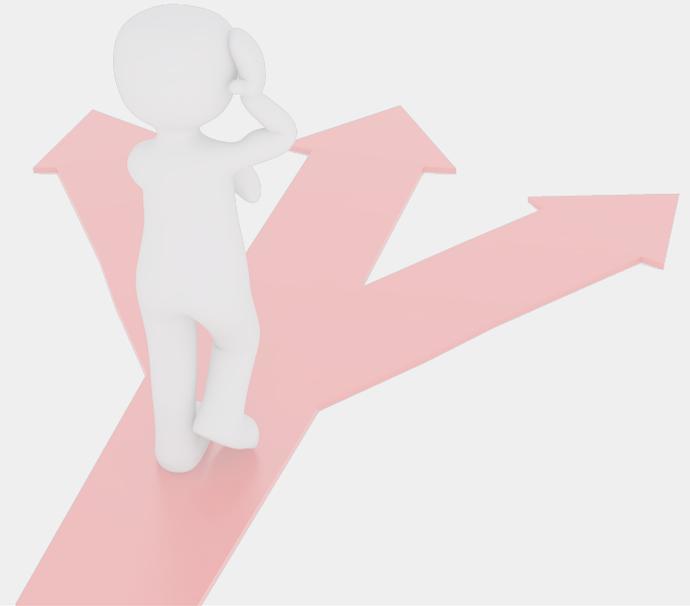


Detektion - Vorhersage - Abgleich - Annotation

Motivation - Ziel - Aufgaben

Wie kann dies umgesetzt werden?

- System zur Steuerung aller notwendigen Prozesse
 - Bildvorbereitung
 - Bildauswertung (Detektion und Vorhersage)
 - Abgleich mit einer Datenbasis
 - Hinzufügen visueller Annotation
- Konzept zum Halten und Nutzen einer Datenbasis
- Konzept zum Einbinden verschiedener neuronaler Netze
⇒ **es kommen externe Netze zum Einsatz**



University of Applied Sciences

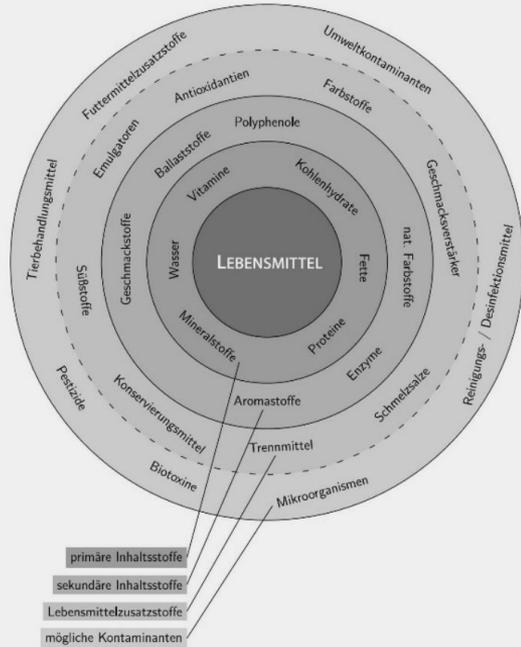
HOCHSCHULE
EMDEN • LEER

Fachbereich Technik
Abteilung Elektrotechnik und Informatik



- Lebensmittelinhaltsstoffe
- Rastergrafiken - RGB - Faltung
- Neuronale Netze
- Python - TensorFlow - Keras

Grundlagen - Lebensmittelinhaltsstoffe



- von Menschen erwartungsgemäß aufnehmbare Stoffe
- lt. Wikipedia* aktuell 361 Stoffe
- Regulierung auf EU und nationaler Basis
- primäre und sekundäre Inhaltsstoffe
- Lebensmittelzusatzstoffe und Kontaminanten

* <https://de.wikipedia.org/w/index.php?oldid=187539415>. Zugriff: 15.05 2019

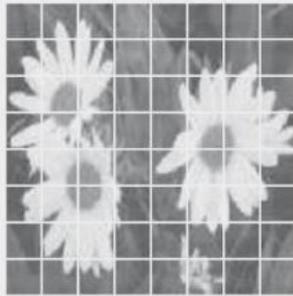
Bild: Systematik der Inhaltsstoffe in Lebensmitteln (Frede 2010, S. 312)

Textextraktion und Einordnung mit Hilfe neuronaler Netze - ein prototypisches System

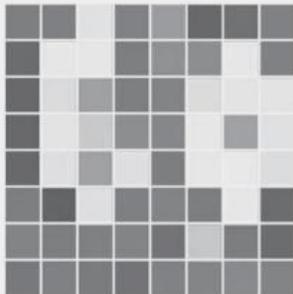
Grundlagen - Rastergrafiken - RGB - Faltung



a



b



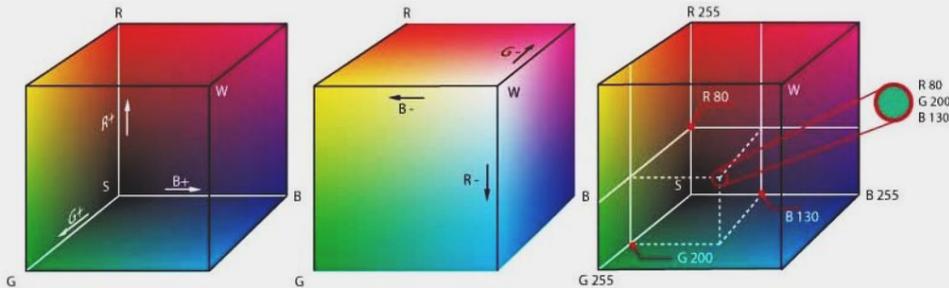
c

18	21	97	12	34	14	13	54
10	95	90	28	35	33	96	53
15	75	19	22	77	75	71	74
22	83	72	25	26	71	15	77
10	83	16	75	42	93	95	72
22	10	72	22	17	19	93	10
21	21	21	21	18	74	12	10
19	21	21	10	23	25	23	23

d

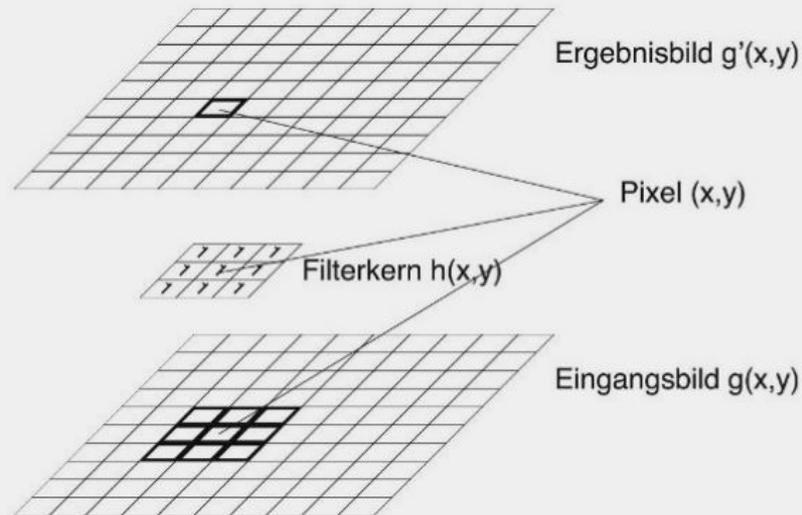
- basiert auf Bildpunkte (Pixel)
- Ergebnis einer Rasterung und Quantisierung
- wird durch Helligkeit und Farbe seiner Pixel beschrieben
- ein verbreitetes Farbmodell ist RGB

Grundlagen - Rastergrafiken - RGB - Faltung



- basiert auf das Verhältnis von Primärfarben (Rot, Grün und Blau)
- $(0, 0, 0) \Rightarrow$ schwarz
 $(255, 255, 255) \Rightarrow$ weiß
- Gleiche Farbwerte ergeben Grau

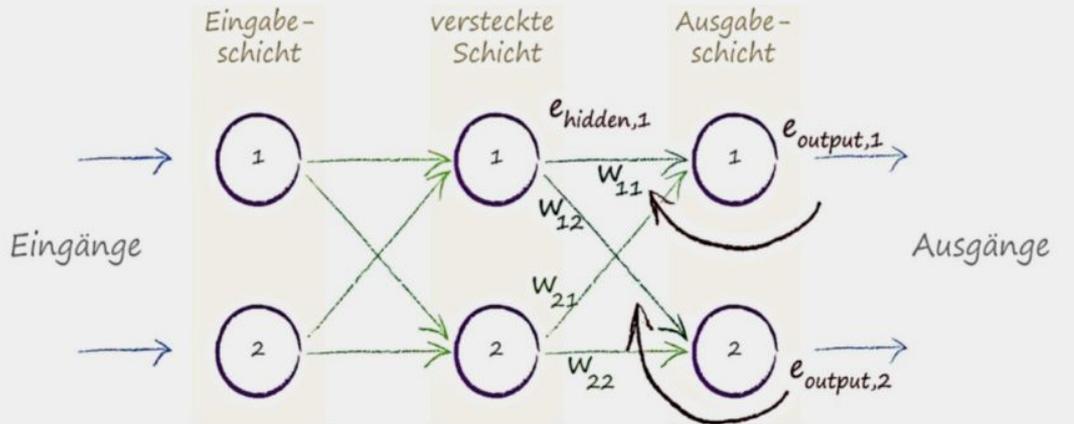
Grundlagen - Rastergrafiken - RGB - Faltung



- fasst Pixel zusammen
- Filterkern und Algorithmus bestimmen die Art und Weise
- Filterkern wandert hierbei über das Eingangsbild
- bei neuronalen Netzen agieren Werte des Filterkerns als Gewichte
⇒ **nicht statisch**

Grundlagen - Neuronale Netze

Aufbau



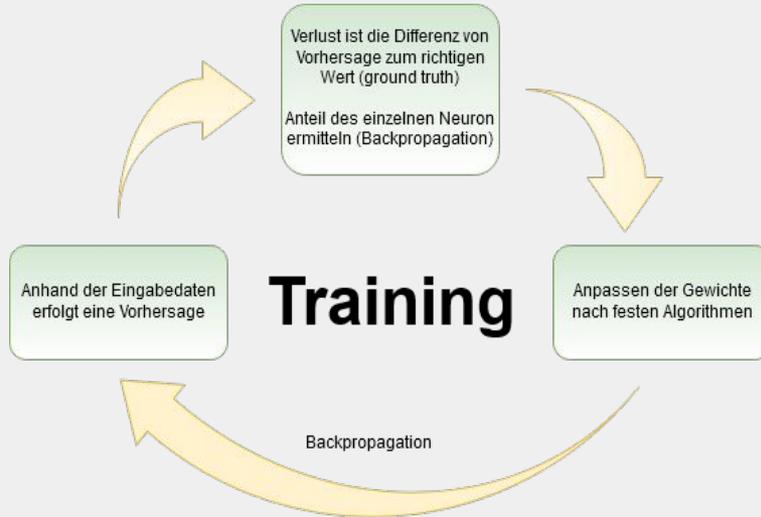
- bestehen aus verbundenen Neuronen
- Neuronen gleicher Tiefe bilden Schichten oder Layer
- Eingabe- und Ausgabeschicht definieren Anfang und Ende
- Dazwischen nicht direkt ansprechbare versteckte Schichten / Layer
- Gewichte und Aktivierungsfunktionen bestimmen den Signalfluss

Bild: Prinzip Backpropagation (Rashid 2017, S. 65)

Textextraktion und Einordnung mit Hilfe neuronaler Netze - ein prototypisches System

Grundlagen - Neuronale Netze

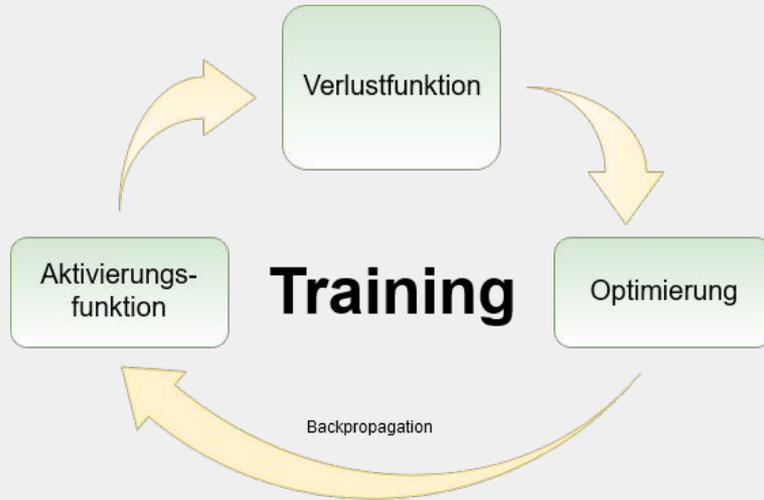
Aktivierung - Verlust - Optimierung



- Training kann als Regelkreis aufgefasst werden
- Eingabedaten führen zu Vorhersagen
- Differenz zwischen Vorhersage und wahrer Aussage ist als Verlust definiert
- Minimierung durch Anpassung der Gewichte

Grundlagen - Neuronale Netze

Aktivierung - Verlust - Optimierung

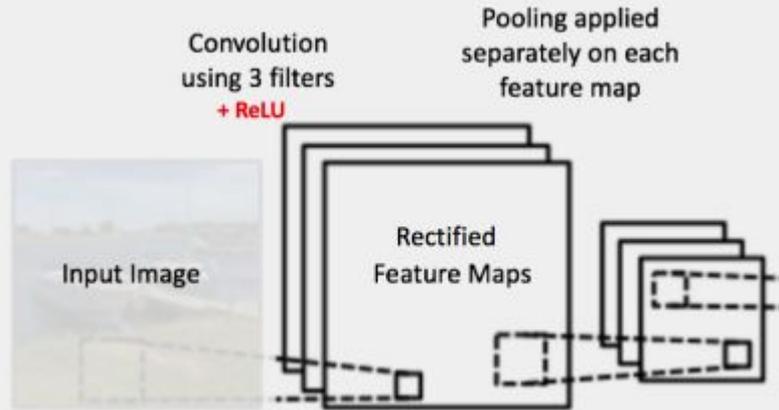


- drei Designkriterien
- Aktivierungsfunktion steuert die charakteristik der Neuronen
- Verlustfunktion beschreibt den Verlust funktional
- Optimierung als Minimierung der Verlustfunktion durch Gewichts-anpassung

Aktivierung - Verlust - Optimierung definieren den Erfolg

Grundlagen - Neuronale Netze

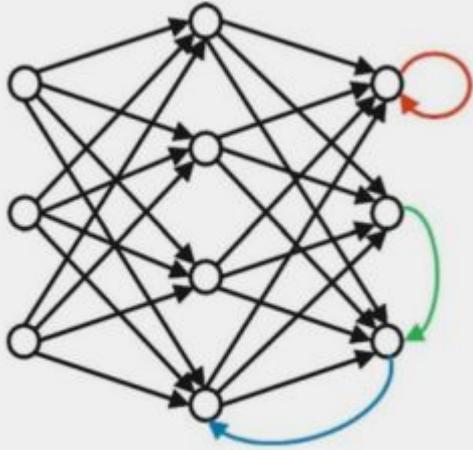
Faltungslayer



- falten Eingabedaten mit einem oder mehreren Filterkernen
- Werte des Filterkerns entsprechen veränderlichen Gewichten
- es entsteht eine oder mehrere neue Schichten (feature maps)
- ermöglicht das Erkennen struktureller Merkmale in Rastergrafiken

Grundlagen - Neuronale Netze

Rückgekoppelte Netze



- Rekurrente Netze (RNN) kennen nicht nur den aktuellen Zustand zur Zeit t
- daneben ist der Zustand zur Zeit $t-1$ verfügbar
- ermöglicht ein Erkennen zeitlicher Merkmale
- Rückkopplungen können direkt (rot), indirekt (blau) oder seitlich (grün) stattfinden

Grundlagen - Entwicklungsumgebung



- Python Version 3.6
- Keras Version 2.2 als “das” Framework für TensorFlow
- TensorFlow Version 1.1 als leistungsfähiges Framework zur Entwicklung neuronaler Netze

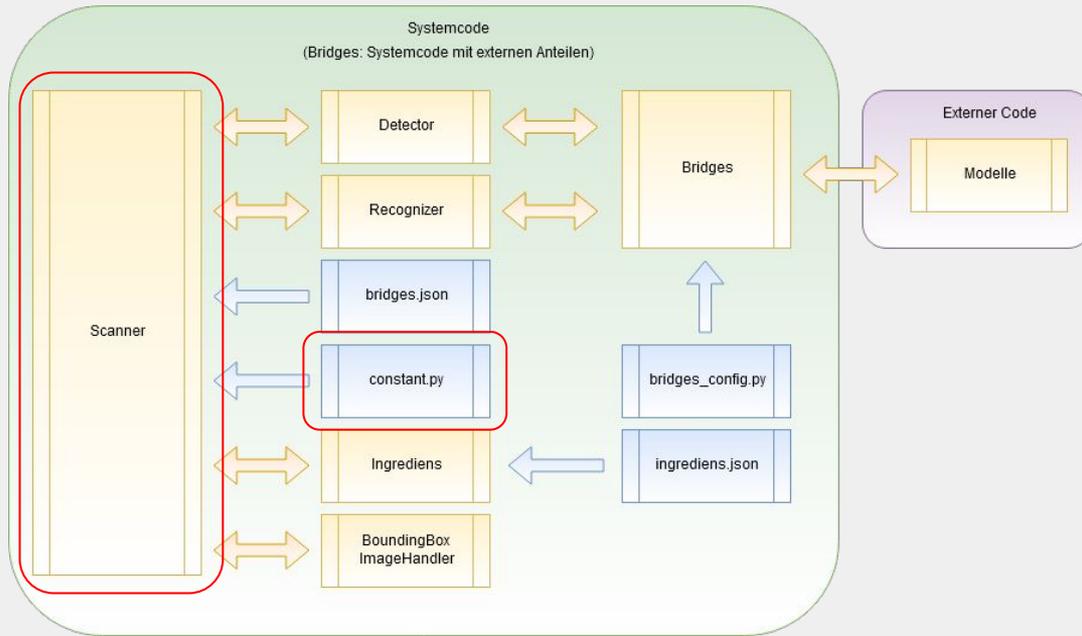
...neben weiteren etablierten Bibliotheken

Umsetzung



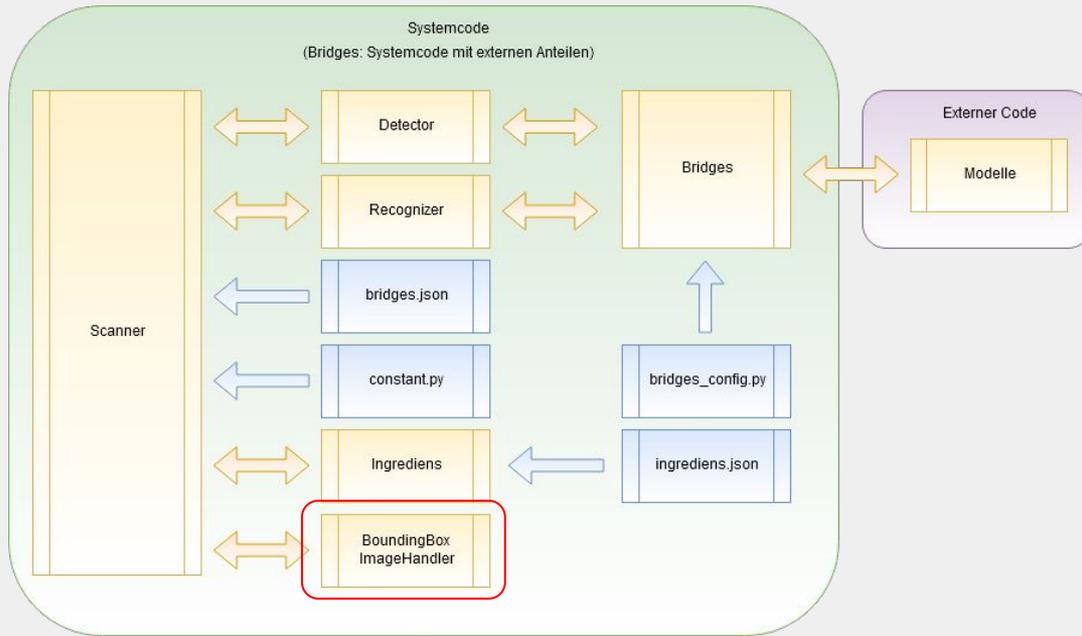
- Gesamtsystem
- Ingrediens
- Detector, Recognizer, Bridge
- OCR - Pipeline

Umsetzung - Gesamtsystem



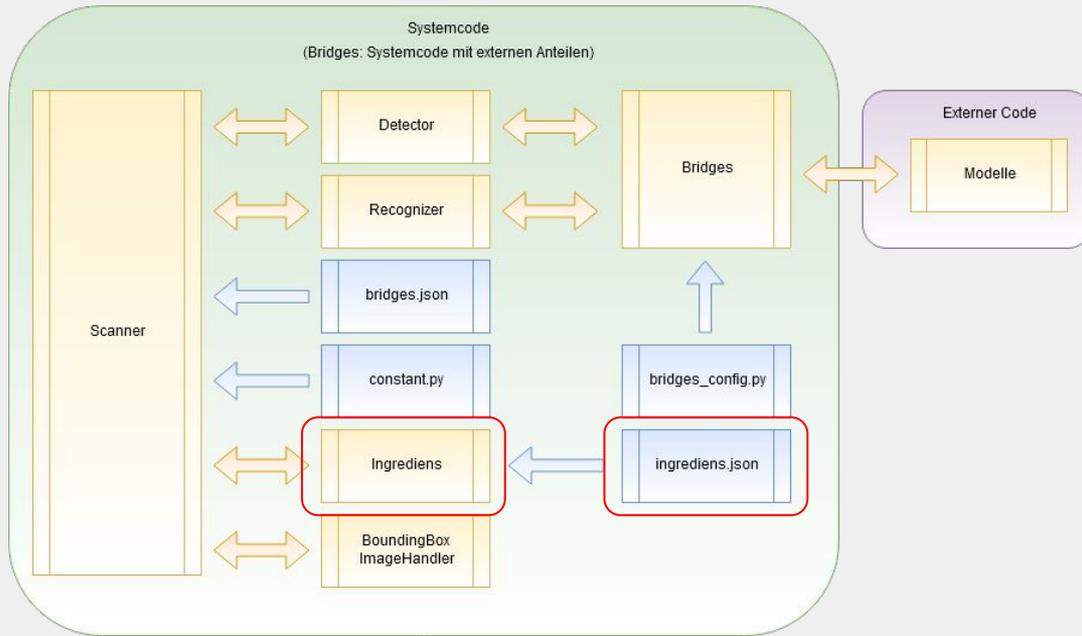
- Scanner als zentrale Komponente
 - constant.py definiert Konstanten

Umsetzung - Gesamtsystem



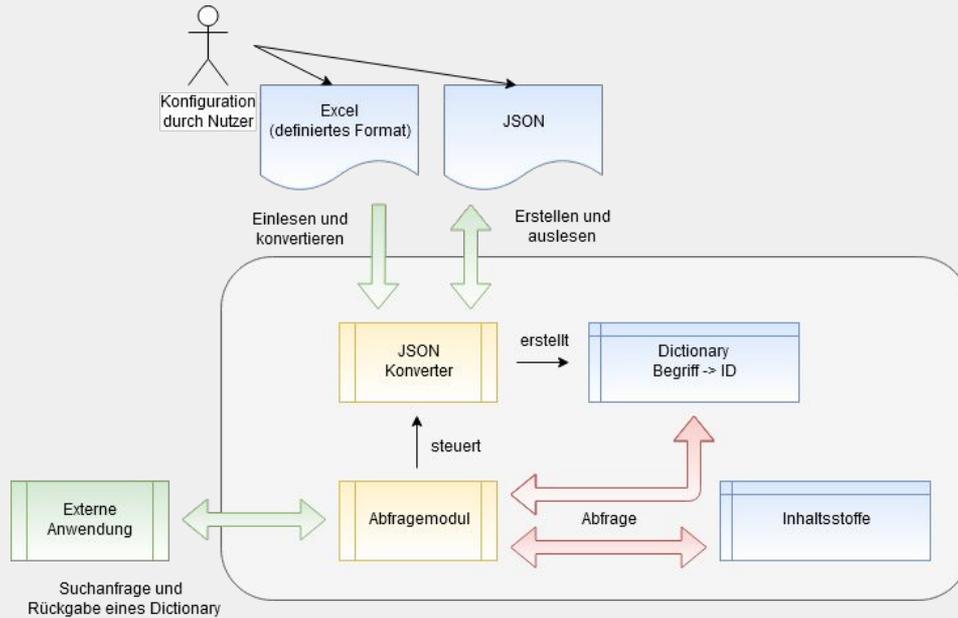
- Scanner als zentrale Komponente
 - `constant.py` definiert Konstanten
- **BoundingBoxImageHandler** als Hilfsbibliothek für Grafik

Umsetzung - Gesamtsystem



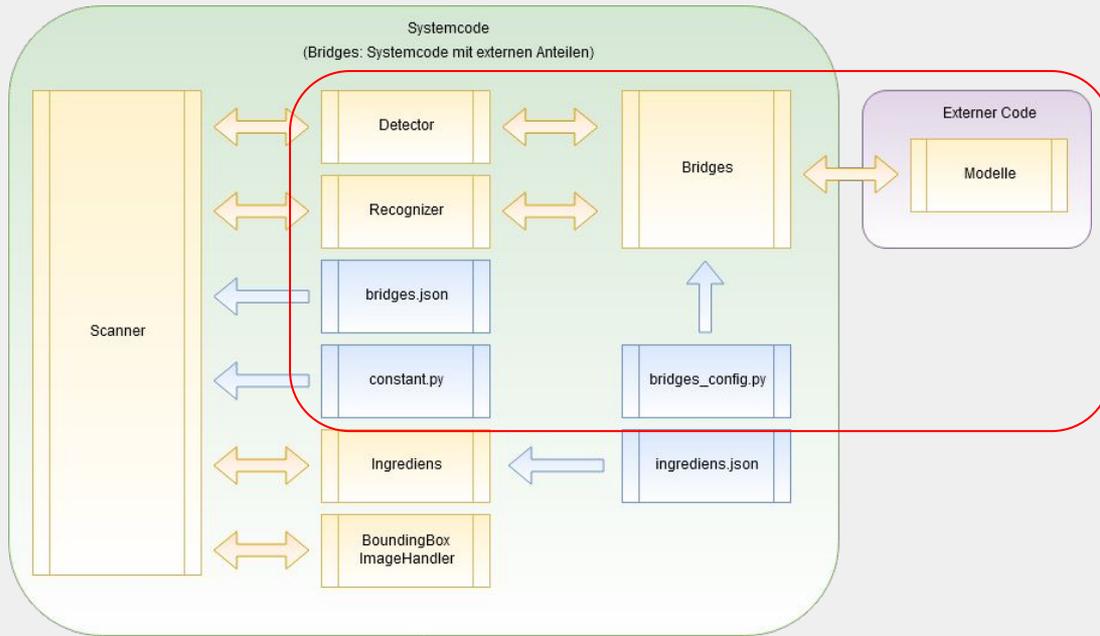
- Scanner als zentrale Komponente
 - `constant.py` definiert Konstanten
- BoundingBoxImageHandler als Hilfsbibliothek für Grafik
- **Ingrediens leistet semantische Zuordnung. Ingrediens.json fungiert als Datenbasis**

Umsetzung - Ingrediens



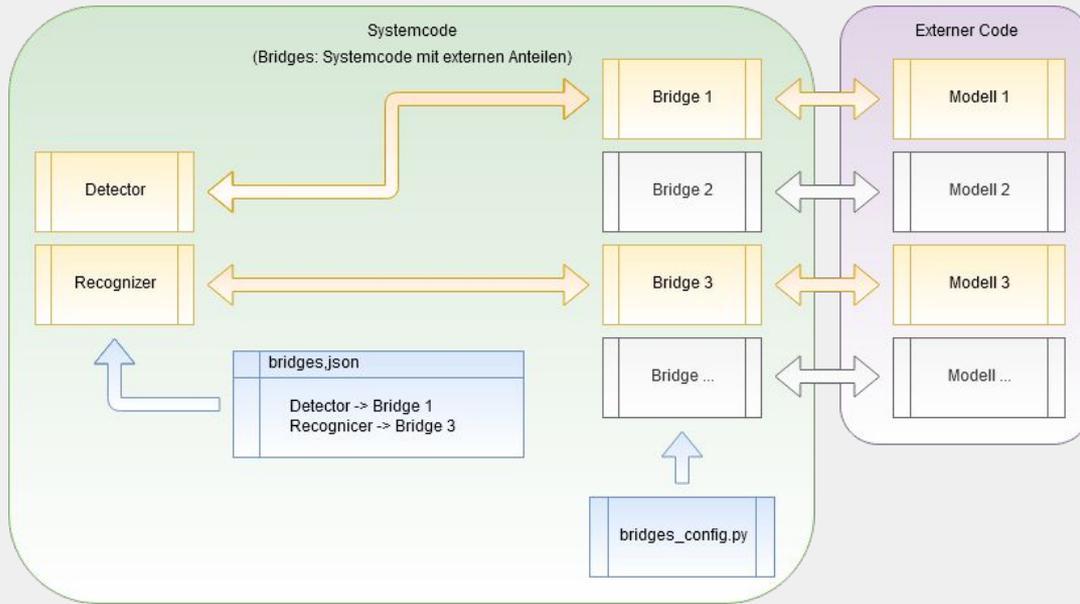
- optionales Excelfile kann in JSON-Datei konvertiert werden
- Datenbasis als JSON-Datei erzeugt Dictionary und Inhaltsstoffe
- Externe Anwendung stellt Anfrage
- Abgleich mit einem Dictionary liefert ID
- Inhaltsstoff über ID verfügbar

Umsetzung - Gesamtsystem



- Scanner als zentrale Komponente
 - `constant.py` definiert Konstanten
- `BoundingBoxImageHandler` als Hilfsbibliothek für Grafik
- `Ingrediens` leistet semantische Zuordnung. `Ingrediens.json` fungiert als Datenbasis
- **Detector und Recognizer leisten die Texterkennung**

Umsetzung - Detector, Recognizer, Bridge



- Detector und Recognizer detektieren und sagen Text voraus
- beide agieren als Fassade zu externen Netzen
- Bridges vermitteln und können externen Code enthalten
- werden durch bridges.json konfiguriert
- bridges_config.py als zentraler Speicher

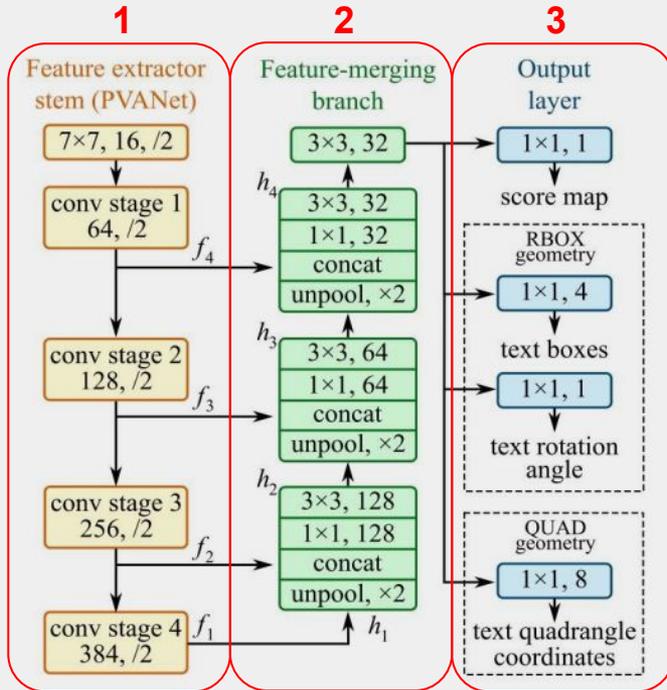
Netzauswahl



- EAST
- CRNN
- CTC

Netzauswahl - EAST (2017) + CRNN mit CTC (2015)

EAST - An Efficient and Accurate Scene Text Detector



1. reiner Textdetektor auf Basis eines Faltungsnetzes (hier PVANET)
2. zwischen Faltungen werden feature maps ausgelesen und gemerged
3. am Ausgang werden die Daten ausgewertet.

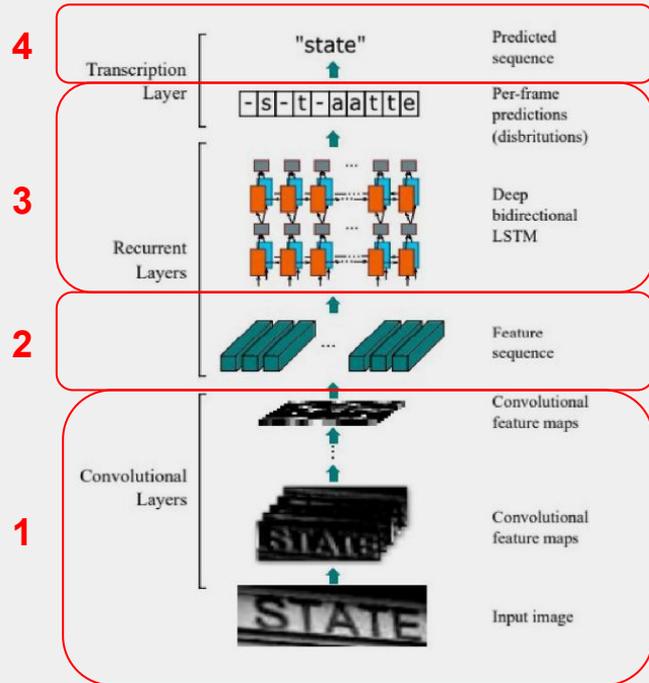
⇒ Textregionen als Liste von bounding boxes ("one shot" Detektor)

Bild: Struktur eines EAST Netzwerkes auf Basis eines PVANET (Zhou et al. 10.07.2017, S. 4323)

Textextraktion und Einordnung mit Hilfe neuronaler Netze - ein prototypisches System

Netzauswahl - EAST (2017) + CRNN mit CTC (2015)

CRNN - Convolutional-Recurrent Neural Network



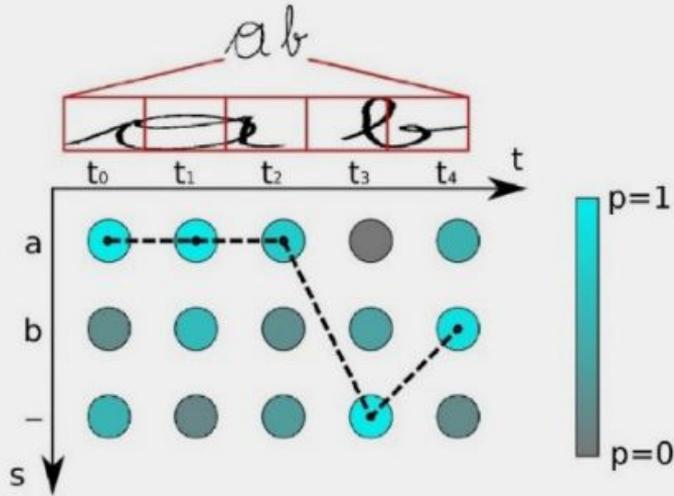
1. ein Faltungsnetz erzeugt feature maps auf Basis einer Rastergrafik
2. durch sequentielle Filterbewegung entsteht eine zeitliche Reihenfolge gefundener features
3. ein rekurrentes Netz wertet die Sequenzen aus
4. das Ergebnis wird mit Hilfe von CTC vorhergesagt

Bild: Struktur eines CRNN zur Texterkennung (Shperber 2019)

Textextraktion und Einordnung mit Hilfe neuronaler Netze - ein prototypisches System

Netzauswahl - EAST (2017) + CRNN mit CTC (2015)

CTC - Connectionist Temporal Classification

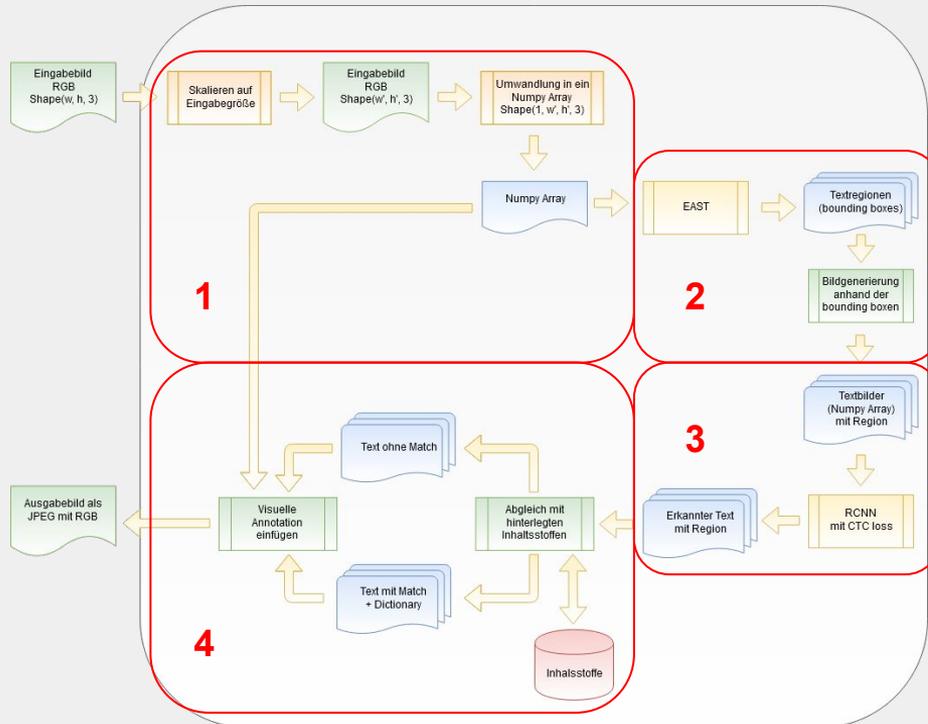


- Sequenz von Ausgaben bilden eine Matrix mit Wahrscheinlichkeiten
- zu jeder Zeit t existiert eine Vorhersage (höchste Wahrscheinlichkeit)
- hierdurch wird ein Graph definiert
- es wird der wahrscheinlichste Graph für ein Ereignis gesucht
- TensorFlow liefert Infrastruktur

Bild: Ausgabematrix als Ergebnis einer Faltung (Scheidl 2018)

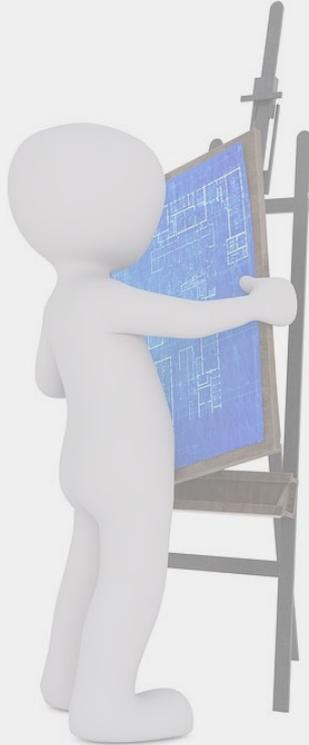
Textextraktion und Einordnung mit Hilfe neuronaler Netze - ein prototypisches System

OCR-Pipeline



1. Eingabebild wird eingelesen, skaliert und in ein NumPy umgewandelt
2. das EAST Netz detektiert Text und ermöglicht Textbilder mit Position
3. enthaltene Text wird mit dem RCNN mit CTC-Loss vorhergesagt
4. Vorhersage ermöglicht Abgleich und visuelle Annotation

Evaluierung



- Problemfelder
- Testbilder

Evaluierung - Problemfelder

Benzoessäure

- **nicht korrekt detektierte Textgrenze**
- nicht korrekt erkannter Text
- nicht korrekter Abgleich

Evaluierung - Problemfelder

Benzooesäure

E 120

- nicht korrekt detektierte Textgrenze
- **nicht korrekt erkannter Text**
- nicht korrekter Abgleich

Evaluierung - Problemfelder

Benzooesäure

E 120

E 150 a, c, d

- nicht korrekt detektierte Textgrenze
- nicht korrekt erkannter Text
- **nicht korrekter Abgleich**

Evaluierung - Testbilder

Detektion und Erkennung von Texten



- definierte Testbilder ermöglichen Vergleichbarkeit
- gleiche Sätze
- verschiedene Hintergründe
- verschiedene Schriftkanten
- verschiedener Verzerrungen

Evaluierung - Testbilder

Detektion und Erkennung von Sonderzeichen

1 2 3 4 5 6 7 8 9 0
ä ö ü Ä Ö Ü
! " § \$ % & / () = ?
E 999
E 999
E-999
E - 999

1 2 3 4 5 6 7 8 9 0
ä ö ü Ä Ö Ü
! " § \$ % & / () = ?
E 999
E 999
E-999
E - 999

- problematische Zeichen
 - Buchstaben
 - Sonderzeichen
 - E-Nummern



⇒ **Detektion und Vorhersage des enthaltenen Textes**

Evaluierung - Testbilder

Erkennung von Text und Sonderzeichen

Der rote Ball ist rund

E 999



1 2 3 4 5 6 7 8 9 0

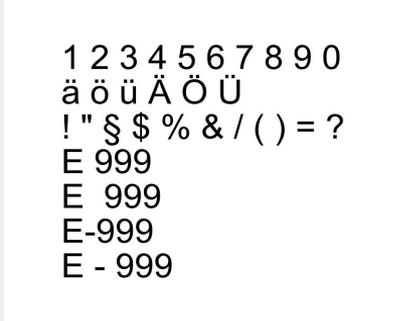


- zusätzlich Textbilder mit definierten Text
- verschiedene Hintergründe
- verschiedene Schriftkanten
- teilweise verzerrt

⇒ **Vorhersage des enthaltenen Textes**

Evaluierung - Testbilder

Verwendung

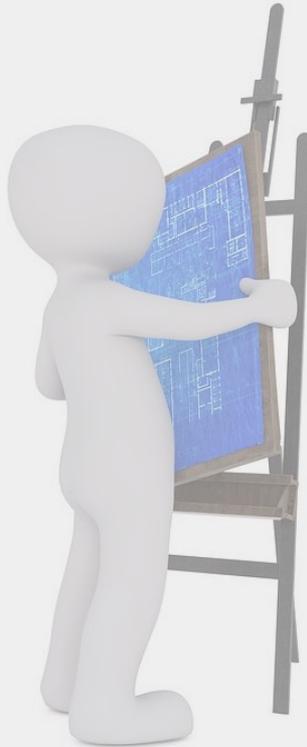


- fest implementierte Methoden
- einfach anzupassen
- einfache Durchführung

Der rote Ball ist rund

⇒ **standardisierte und wiederholbare Tests**

Evaluierung - Ergebnis



- Abgleich von Begriffen
- Detektion und Erkennen
- Erkennen
- Ergebnis

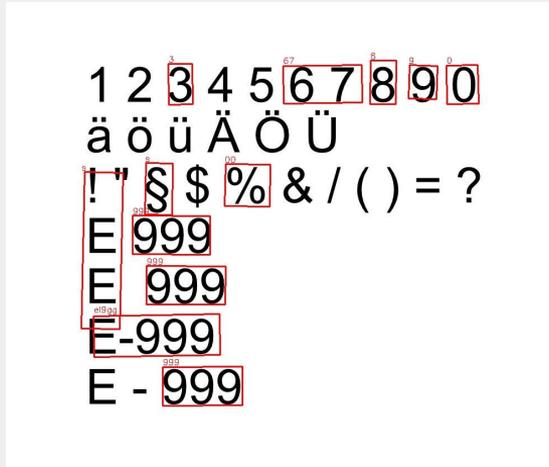
Evaluierung - Ergebnis Abgleich von Begriffen

```
E-160e evaluate to True stands for E 160e - Beta-apo-8-Carotinal
E - 160e evaluate to True stands for E 160e - Beta-apo-8-Carotinal
e-160e evaluate to True stands for E 160e - Beta-apo-8-Carotinal
e - 160e evaluate to True stands for E 160e - Beta-apo-8-Carotinal
E160 e evaluate to True stands for E 160e - Beta-apo-8-Carotinal
```

- Systematisches Abfrage von Begriffen
- werden Begriffe korrekt erkannt
- bei verschiedene Schreibweise
- in verschiedene Datenfelder
 - ⇒ E-Nummer
 - ⇒ Namen
 - ⇒ Schlüsselwörter

Evaluierung - Ergebnis Detektion und Erkennung

(am Beispiel weißer Hintergrund, scharfer Rand)

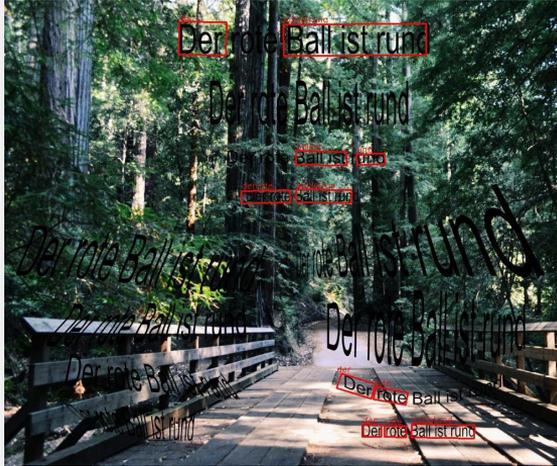


- wie akkurat ist die Detektion
- wie akkurat ist die Erkennung
- im Zusammenspiel

⇒ gute Detektion und Vorhersage von Text

Evaluierung - Ergebnis Detektion und Erkennung

(am Beispiel Bildhintergrund, scharfer Rand)



⇒ schlechte Detektion und Vorhersage bei Bildhintergrund

Evaluierung - Ergebnis Erkennung

(am Beispiel waagerechter Text ohne Verzerrung)



001_forrest_sharp_blur_derrotebalistung.jpg



001_forrest_sharp_derrotebalistung.jpg



001_sharp_blur_derroteballistru.jpg



001_sharp_derroteballistru.jpg

- werden Buchstaben korrekt erkannt
- werden Sonderzeichen korrekt erkannt
- werden Leerzeichen korrekt erkannt
- bei gegebenen Ausschnitt

Evaluierung - Zusammenfassung

allgemein

- **strukturierter Workflow zum Testen**
- **Wiederholbarkeit und Vergleichbarkeit der gewonnenen Ergebnisse für verschiedene neuronale Netze**
- **leichte Erweiterung und Anpassung an verschiedenen Szenarien möglich**
- **Ergebnisse für die aktuelle Konfiguration**
 - Abgleich gefundener Begriffe arbeitet wie erwartet
 - Texte werden erkannt und enthaltene Begriffe vorhergesagt
 - jedoch Probleme bei Verzerrung und nicht weißen Hintergrund



Evaluierung - Zusammenfassung

für die aktuelle Konfiguration

- strukturierter Workflow zum Testen
- Wiederholbarkeit und Vergleichbarkeit der gewonnenen Ergebnisse für verschiedene neuronale Netze
- leichte Erweiterung und Anpassung an verschiedenen Szenarien möglich
- **Ergebnisse für die aktuelle Konfiguration**
 - Abgleich gefundener Begriffe arbeitet wie erwartet
 - Texte werden erkannt und enthaltene Begriffe vorhergesagt
 - jedoch Probleme bei Verzerrung und nicht weißen Hintergrund



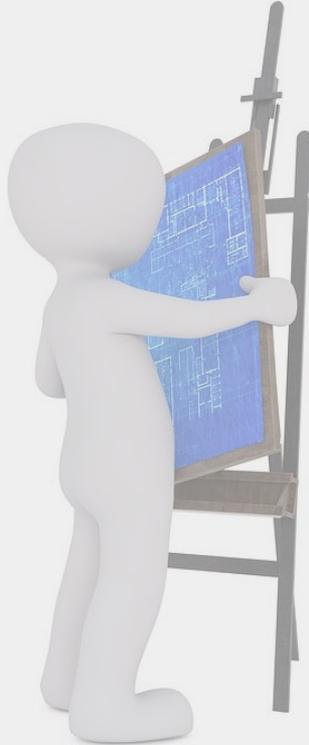
Evaluierung - Zusammenfassung

für die aktuelle Konfiguration

- strukturierter Workflow zum Testen
- Wiederholbarkeit und Vergleichbarkeit der gewonnenen Ergebnisse für verschiedene neuronale Netze
- leichte Erweiterung und Anpassung an verschiedenen Szenarien möglich
- **Ergebnisse für die aktuelle Konfiguration**
 - Abgleich gefundener Begriffe arbeitet wie erwartet
 - Texte werden erkannt und enthaltene Begriffe vorhergesagt
 - jedoch Probleme bei Verzerrung und nicht weißen Hintergrund
⇒ **Optimierungsbedarf**



Zusammenfassung



- Fazit
- Ausblick

Zusammenfassung - Fazit

Was wurde erreicht

- **die Aufgabenstellung konnte grundsätzlich erfüllt werden**
 - gut bei horizontalen Texten auf weißem Hintergrund
 - Framework für den Einsatz beliebiger neuronaler Netze
 - Basis für einheitliches und wiederholbares Prüfverfahren
- **konkret eingesetzten Modelle schwächeln**
 - bei farbigen Hintergründen
 - bei verzerrten Textvorkommen
 - bei Sonderzeichen



Zusammenfassung - Fazit

Was wurde erreicht

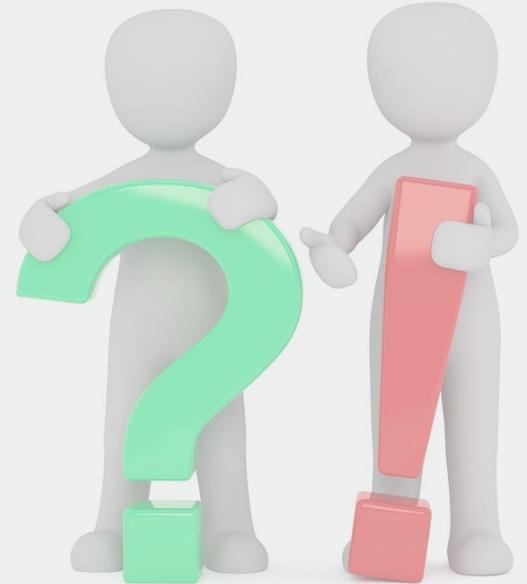
- **die Aufgabenstellung konnte grundsätzlich erfüllt werden**
 - gut bei horizontalen Texten auf weißem Hintergrund
 - Framework für den Einsatz beliebiger neuronaler Netze
 - Basis für einheitliches und wiederholbares Prüfverfahren
- **konkret eingesetzten Modelle schwächeln**
 - bei farbigen Hintergründen
 - bei verzehrten Textvorkommen
 - bei Sonderzeichen



Zusammenfassung - Ausblick

Wie könnte es weitergehen?

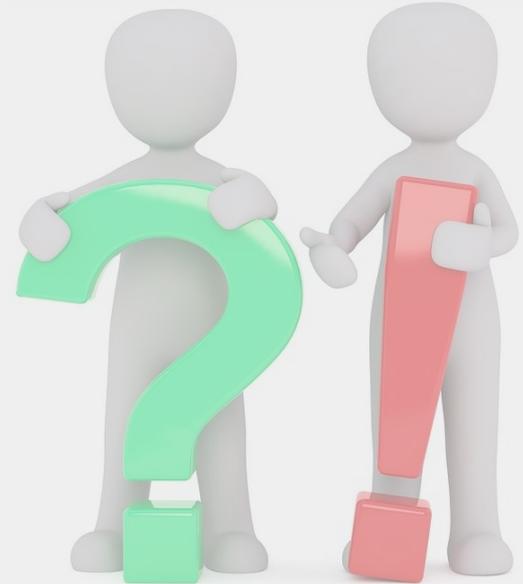
- **Alternative neuronaler Netze**
 - für die Detektion von Textbereichen
 - für die Vorhersage des Textes
- **Anpassung durch finetuning**
 - anhand besserer Beispielbilder
 - anhand deutscher Wortvorkommen
- **Vorbereiten der Eingabedaten**
 - Bessere Auflösungen
 - Kontrasterhöhung
 - Hervorhebung von Kanten



Zusammenfassung - Ausblick

Wie könnte es weitergehen?

- **Alternative neuronaler Netze**
 - für die Detektion von Textbereichen
 - für die Vorhersage des Textes
- **Anpassung durch finetuning**
 - anhand besserer Beispielbilder
 - anhand deutscher Wortvorkommen
- **Vorbereiten der Eingabedaten**
 - Bessere Auflösungen
 - Kontrasterhöhung
 - Hervorhebung von Kanten



University of Applied Sciences

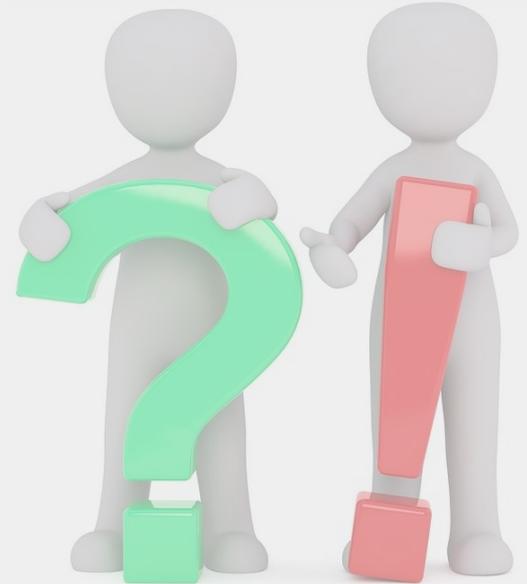
HOCHSCHULE
EMDEN • LEER

Fachbereich Technik
Abteilung Elektrotechnik und Informatik

Zusammenfassung - Ausblick

Wie könnte es weitergehen?

- **Alternative neuronaler Netze**
 - für die Detektion von Textbereichen
 - für die Vorhersage des Textes
- **Anpassung durch finetuning**
 - anhand besserer Beispieldaten
 - anhand deutscher Wortvorkommen
- **Vorbereiten der Eingabedaten**
 - Bessere Auflösungen
 - Kontrasterhöhung
 - Hervorhebung von Kanten



University of Applied Sciences

HOCHSCHULE
EMDEN • LEER

Fachbereich Technik
Abteilung Elektrotechnik und Informatik



**Vielen Dank für Ihre
Aufmerksamkeit!**

Haben Sie noch Fragen?

Gerne beantworte ich Ihnen diese vor oder nach
der Vorführung...

Marken und Rechte

Auf Grund der besseren Lesbarkeit wurden für geschützte Begriffe, Warennamen, Marken usw. keine Angaben zu den Rechten Dritter gemacht. Die Verwendung in dieser Arbeit berechtigt daher nicht zu der Annahme, dass diese frei von Rechten Dritter sind.

Bildnachweis mit Bilder mit Quellenangabe

siehe Literatur- und Quellenverzeichnis Folie 56

Bild: Systematik der Inhaltsstoffe in Lebensmitteln (Frede 2010, S. 312)	(Folie 08)
Bild: Prinzip Digitalisierung (Erhardt 2008, S. 80)	(Folie 09)
Bild: RGB – Farbwürfel (Wikipedia-Autoren 14.11.2004)	(Folie 10)
Bild: Prinzip der Faltung (Erhardt 2008, S. 154)	(Folie 11)
Bild: Prinzip Backpropagation (Rashid 2017, S. 65)	(Folie 12)
Bild: Prinzip Faltung (Karn 2016, letzter Zugriff 18.09.2019)	(Folie 15)
Bild: Prinzip RNN (Dörn 2018, S. 127)	(Folie 16)
Bild: Struktur eines EAST Netzwerkes auf Basis eines PVANET (Zhou et al. 10.07.2017, S. 4323)	(Folie 26)
Bild: Struktur eines CRNN zur Texterkennung (Shperber 2019)	(Folie 27)
Bild: Ausgabematrix als Ergebnis einer Faltung (Scheidl 2018)	(Folie 28)

Bildnachweis ohne Quellenangabe

Sonstige Bilder

Für alle Bilder der Plattform Pixabay (<https://pixabay.com/>) gelten deren Lizenzbedingungen. Diese kann unter <https://pixabay.com/de/service/terms/#license> eingesehen werden.

https://pixabay.com/de/checkliste-liste-%C3%BCberpr%C3%BCfen-marke-1919328/	(Folie 2)
https://pixabay.com/de/richtung-weg-entscheidung-ziel-2320124/	(Folie 3, 4, 5, 6)
https://pixabay.com/de/staffelei-schulung-bildung-training-2714167/	(Folie 7, 18, 25, 30, 38, 46)
https://pixabay.com/de/tafel-staffelei-architekt-ingenieur-2714168/	(Folie 47, 48)
https://pixabay.com/de/fragezeichen-frage-hilfe-antwort-2314115/	(Folie 49, 50, 51)
https://pixabay.com/de/fragezeichen-frage-antwort-1019993/	(Folie 52)
https://pixabay.com/de/illustrations/abw%C3%A4gen-plus-minus-waage-2856321/	(Folie 43, 44, 45)

Bildnachweis ohne Quellenangabe

Sonstige Bilder

[https://en.wikipedia.org/wiki/Python_\(programming_language\)#/media/File:Python_logo_and_word_mark.svg](https://en.wikipedia.org/wiki/Python_(programming_language)#/media/File:Python_logo_and_word_mark.svg) (Folie 17)

By www.python.org - <https://www.python.org/community/logos/>, GPL,
<https://commons.wikimedia.org/w/index.php?curid=34991637>

https://en.wikipedia.org/wiki/Keras#/media/File:Keras_Logo.jpg (Folie 17)

By Francois Chollet - Keras Logo, Public Domain, <https://commons.wikimedia.org/w/index.php?curid=51518645>

<https://de.wikipedia.org/wiki/TensorFlow#/media/File:TensorFlowLogo.svg> (Folie 17)

By TensorFlow - vectors combined, edited - Begoon -
<https://github.com/tensorflow/tensorflow><https://github.com/valohai/ml-logos/blob/master/tensorflow-text.svg><https://github.com/m/valohai/ml-logos/blob/master/tensorflow-tf.svg>, Apache License 2.0,
<https://commons.wikimedia.org/w/index.php?curid=65268375>

Für alle sonstigen Abbildungen liegen die Rechte beim Autor

Literatur- und Quellenverzeichnis

Dörn S (2018) Programmieren für Ingenieure und Naturwissenschaftler. Springer Berlin Heidelberg, Berlin, Heidelberg

Erhardt A (2008) Einführung in die Digitale Bildverarbeitung; Grundlagen, Systeme und Anwendungen. Vieweg+Teubner / GWV Fachverlage GmbH Wiesbaden, Wiesbaden

Frede W (2010) Handbuch für Lebensmittelchemiker. Springer Berlin Heidelberg, Berlin, Heidelberg

Karn Ujjwal (11.08.2016) An Intuitive Explanation of Convolutional Neural Networks. <https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets>. Zugegriffen: 18. September 2019

Rashid T (2017) Neuronale Netze selbst programmieren; Ein verständlicher Einstieg mit Python. O'Reilly, Heidelberg

Scheidl H (2018) An Intuitive Explanation of Connectionist Temporal Classification. <https://towardsdatascience.com/intuitively-understanding-connectionist-temporal-classification-3797e43a86c>. Zugegriffen: 31. Juli 2019

Shperber G (2019) A gentle introduction to OCR. <https://towardsdatascience.com/a-gentle-introduction-to-ocr-ee1469a201aa>. Zugegriffen: 29. Juli 2019

Wikipedia-Autoren sV (14.11.2004) Datei:RGB farbwuerfel.jpg. https://de.wikipedia.org/wiki/Datei:RGB_farbwuerfel.jpg. Zugegriffen: 18. Mai 2019

Literatur- und Quellenverzeichnis

Zhou X, Yao C, Wen H, Wang Y, Zhou S, He W, Liang J (10.07.2017) EAST: An Efficient and Accurate Scene Text Detector.
<https://arxiv.org/pdf/1704.03155>. Zugegriffen: 26. August 2019