

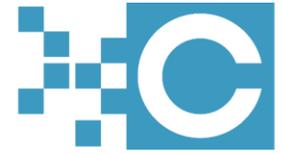


CIRCULAR INFINITY GMBH

Implementierung eines bildverarbeitenden Verfahrens zur quantitativen Bestimmung von Stoffanteilen in siedlungsabfallwirtschaftlichen Stoffströmen

Kevin Höfle, Tobias Althoff, Ralf Holzhauer

Autoren



CIRCULAR INFINITY GMBH



Fh.-Ing Kevin Höfle M.Sc.

- Master in Informationstechnik, Ingenieur der Technischen Informatik
- Maschinelles Lernen / Deep Learning in der circular infinity GmbH
- Doktorand an der Hochschule Mannheim
- Gesellschafter der circular infinity GmbH



Tobias Althoff, M.Sc.

- Ingenieur der Entsorgungssystemtechnik
- Zentrum für Recyclingtechnik, Westfälische Hochschule
- Doktorand an der Universität Duisburg-Essen
- Geschäftsführer / Gesellschafter der circular infinity GmbH



Prof. Dr.-Ing. Ralf Holzhauer

- Professor an der Westfälischen Hochschule
- Zentrum für Recyclingtechnik, Westfälische Hochschule
- Wertschöpfungsketten in der Kreislaufwirtschaft
- Gesellschafter der circular infinity GmbH



Inhalt

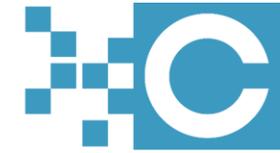
- Einleitung
- Grundlagen
- Methodik
- Ergebnisse
- Fazit & Zukunft



CIRCULAR INFINITY GMBH

Einleitung

Einleitung | Grundlagen | Methodik | Ergebnisse | Fazit & Zukunft



Einleitung

- Sortenreine Siedlungsabfallströme bilden eine Grundlage für die nachhaltige Etablierung einer zirkulären Wertschöpfung
- Durch die Beschreibung der stofflichen Zusammensetzung der Siedlungsabfallströme können Strategien zur Optimierung von Sammelsystemen gestaltet werden



Ziel

- Prüfung, ob es möglich ist unter der Verwendung von Deep Learning eine quantitative Bestimmung von Stoffanteilen in siedlungsabfallwirtschaftlichen Stoffströmen anhand von Bild-Daten zu bestimmen



CIRCULAR INFINITY GMBH

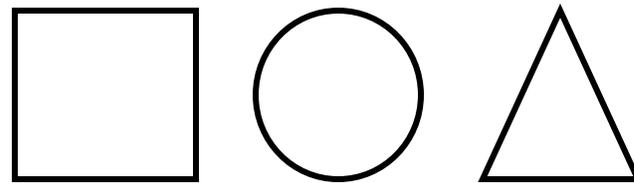
Grundlagen

Einleitung | Grundlagen | Methodik | Ergebnisse | Fazit & Zukunft



Deep Convolutional Neural Networks

- Beispiel an geometrischen Formen

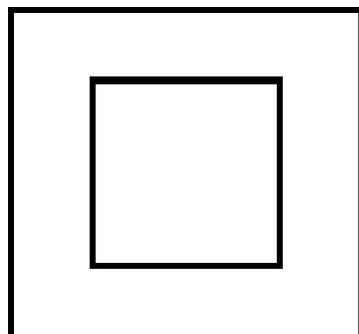


- Erkennung von Viereck, Kreis und Dreieck
- Erkennung über „Faltungen“ → Summe überlappender Flächen

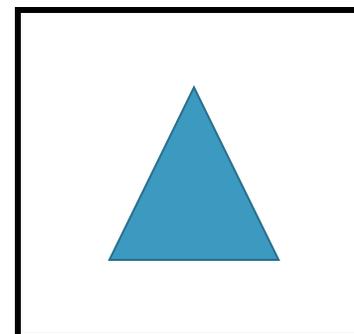
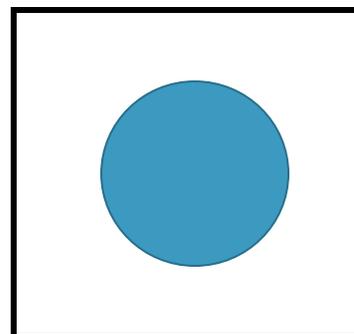
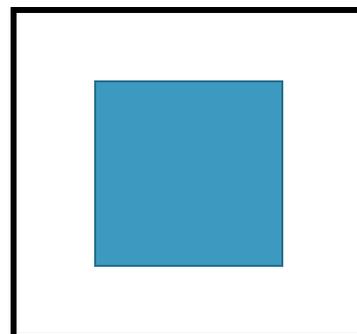


Deep Convolutional Neural Networks

Input – Bild



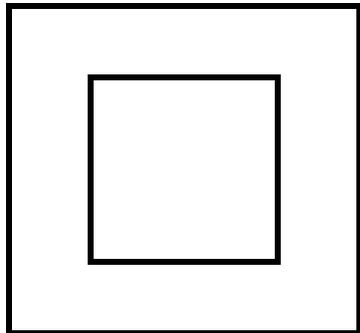
Faltungskerne



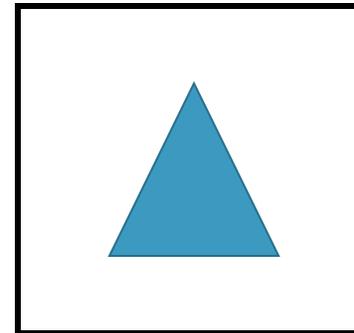
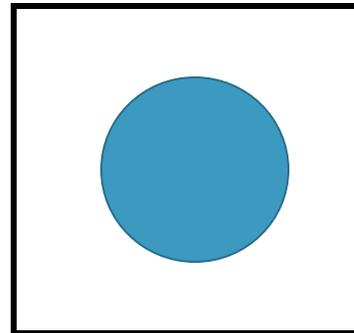
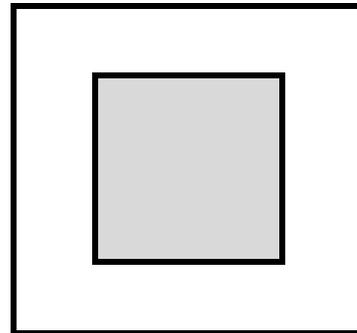


Deep Convolutional Neural Networks

Input – Bild



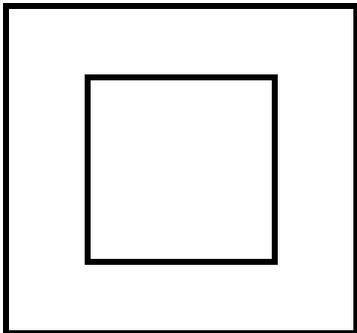
Faltungskerne



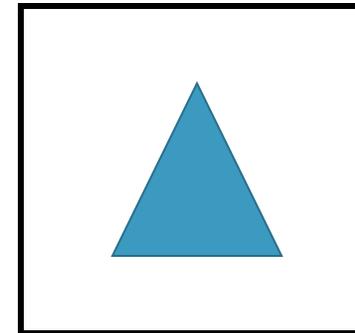
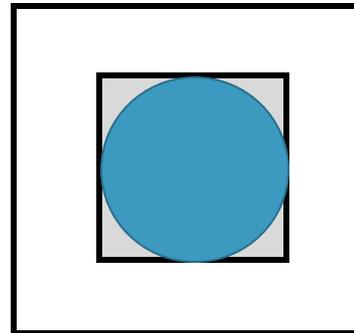
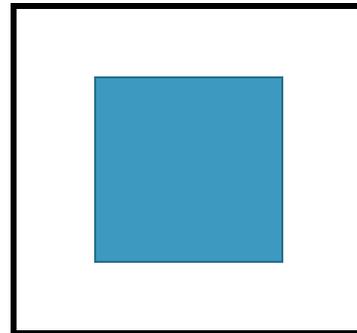
Perfekte
Überlappung

Deep Convolutional Neural Networks

Input – Bild



Faltungskerne

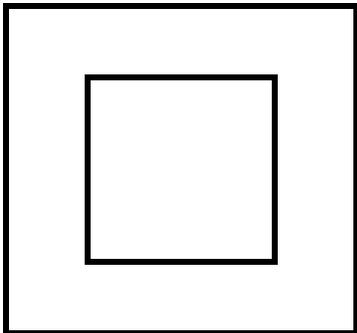


Perfekte
Überlappung

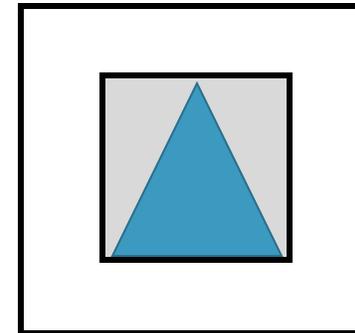
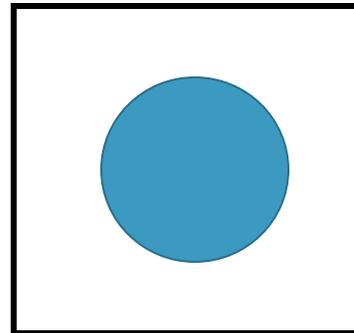
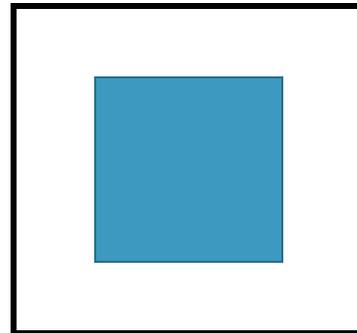
Mittlere
Überlappung

Deep Convolutional Neural Networks

Input – Bild



Faltungskerne



Perfekte
Überlappung

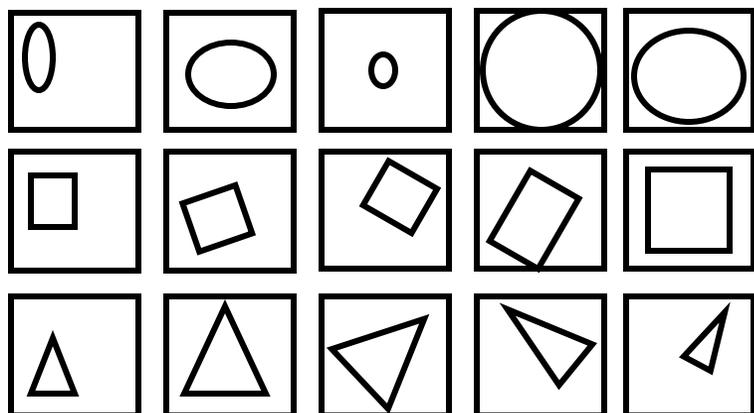
Mittlere
Überlappung

Geringe
Überlappung

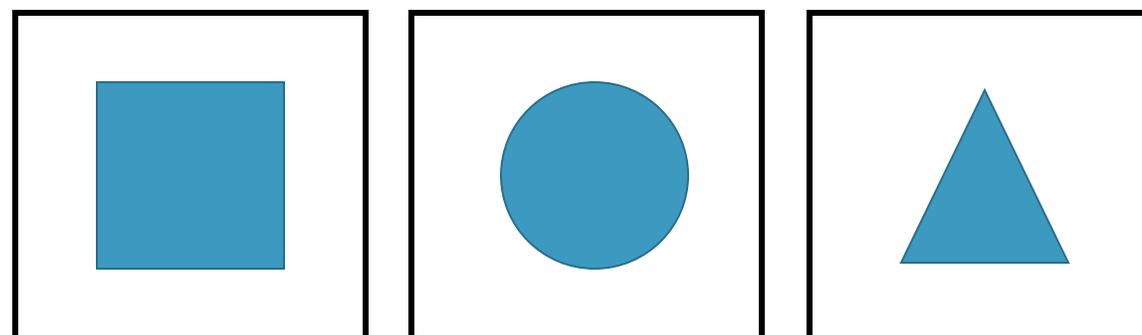


Deep Convolutional Neural Networks

Input – Bilder



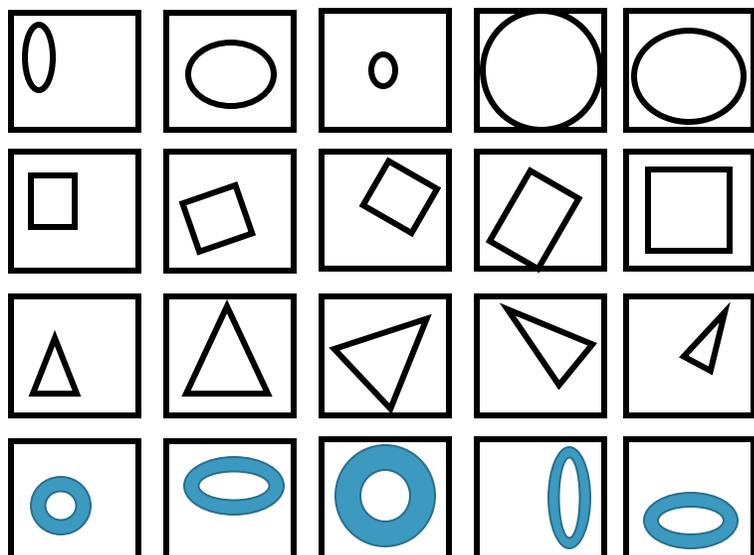
Faltungskerne



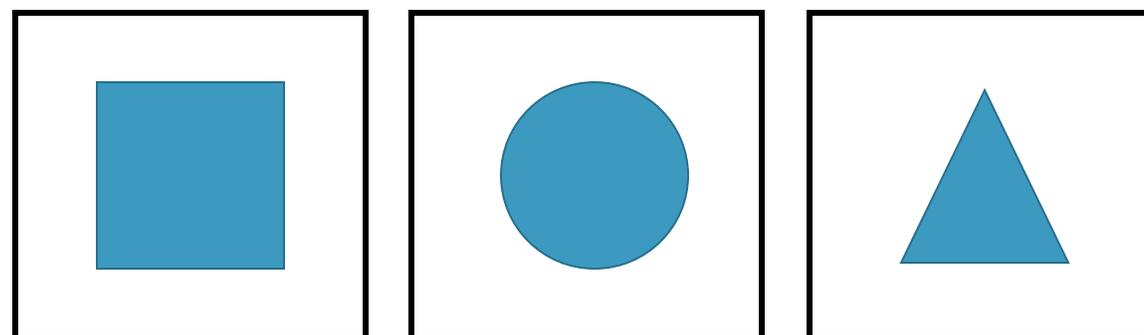


Deep Convolutional Neural Networks

Input – Bilder

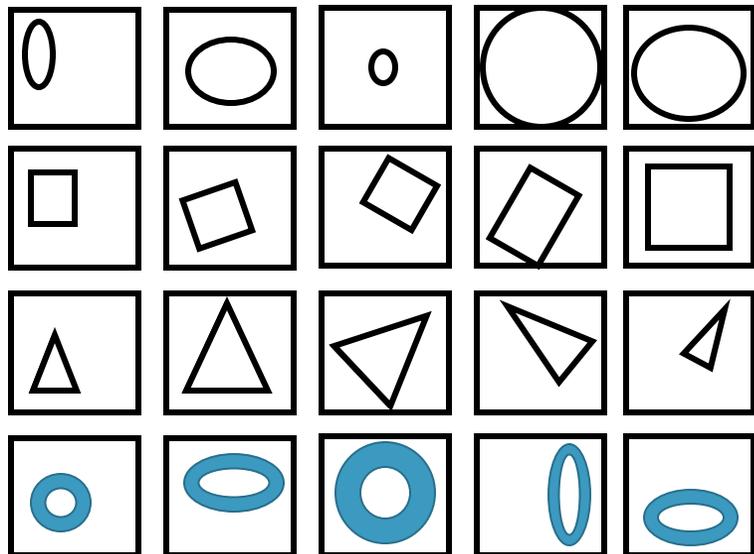


Faltungskerne

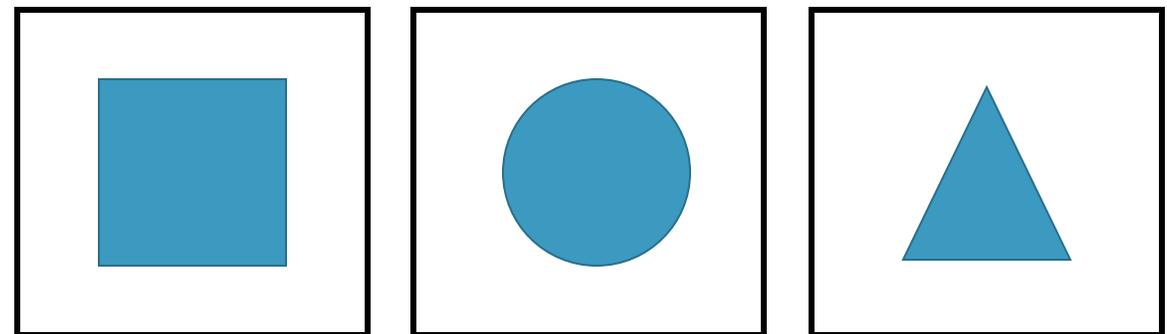


Deep Convolutional Neural Networks

Input – Bilder



Faltungskerne



- Je mehr Varianz ein zu erkennendes Objekt bietet, desto komplexer werden die Faltungskerne
- Besonders schwer bei natürlichen Gegenständen (Mensch, Katze, Hund, Auto, LKW,...)
- Erkennung anhand von Kanten und Kombinationen von Kanten (Kantendetektion)



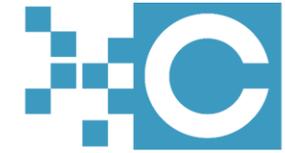
Deep Convolutional Neural Networks

- Faltungskerne sind schwer zu bestimmen, da jede Kombination, die in der Natur vorkommen kann berücksichtigt werden muss
 - Beispiel am Auto:
 - Ausrichtung: oben, unten, vorne, hinten, links, rechts,...
 - Farben: rot, grün, blau, weiß, schwarz, silbern,
 - Beleuchtung: hell, dunkel,.... (witterungsbedingt)
 - „Formen“: Kombi, Limousine, Kleinwagen,....



Deep Convolutional Neural Networks

- Faltungskerne sind schwer zu bestimmen, da jede Kombination, die in der Natur vorkommen kann berücksichtigt werden muss
 - Beispiel am Auto:
 - Ausrichtung: oben, unten, vorne, hinten, links, rechts,...
 - Farben: rot, grün, blau, weiß, schwarz, silbern,
 - Beleuchtung: hell, dunkel,.... (witterungsbedingt)
 - „Formen“: Kombi, Limousine, Kleinwagen,....
- Daten bereitstellen, die das zu erwartende Ergebnis sowie eine möglichst hohe Varianz innerhalb der Klassen aufweisen



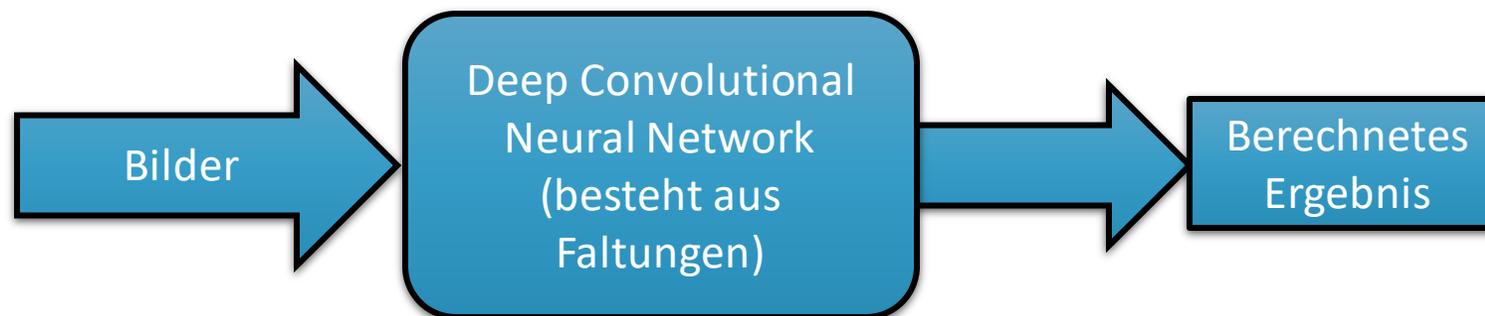
Deep Convolutional Neural Networks

- Faltungskerne sind schwer zu bestimmen,
- ... aber über eine Lernfunktion können Faltungskerne, und sogar Kombinationen von Kernen gelernt werden
- Bilder in Kombination mit dem zu erwartenden Ergebnis können verwendet werden



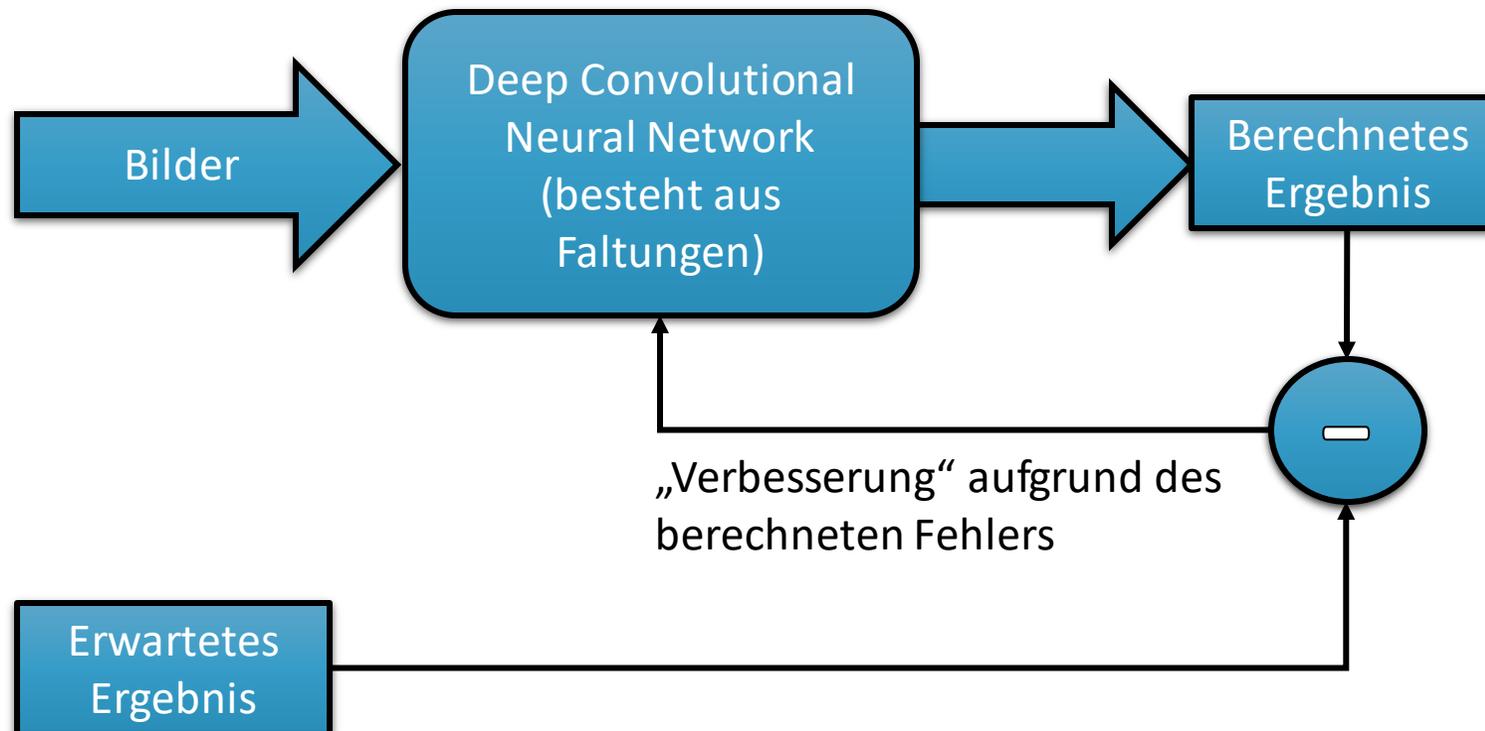
Deep Convolutional Neural Networks

- ... aber über eine Lernfunktion können Faltungskerne, und sogar Kombinationen von Kernen gelernt werden



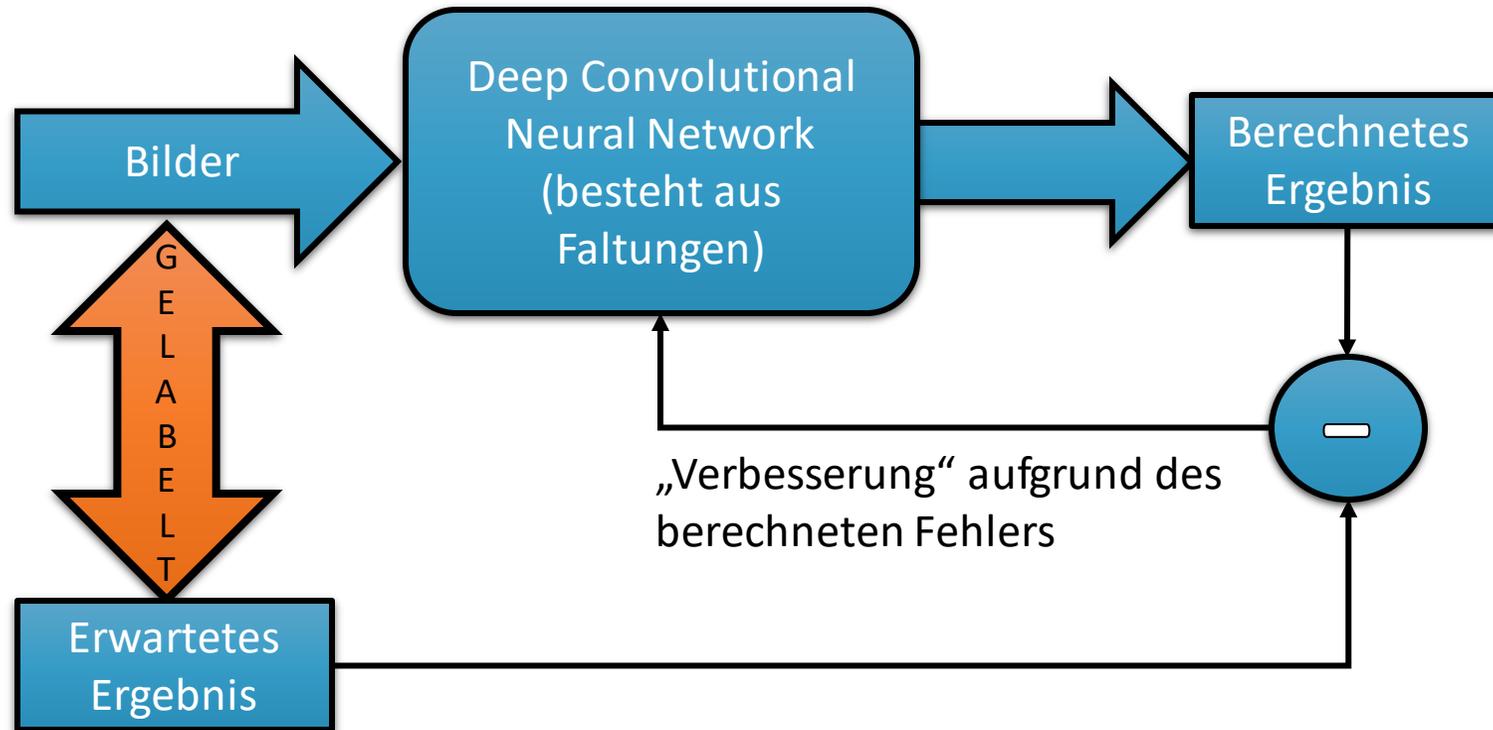
Deep Convolutional Neural Networks

- ... aber über eine Lernfunktion können Faltungskerne, und sogar Kombinationen von Kernen gelernt werden





Deep Convolutional Neural Networks



Vorteile:

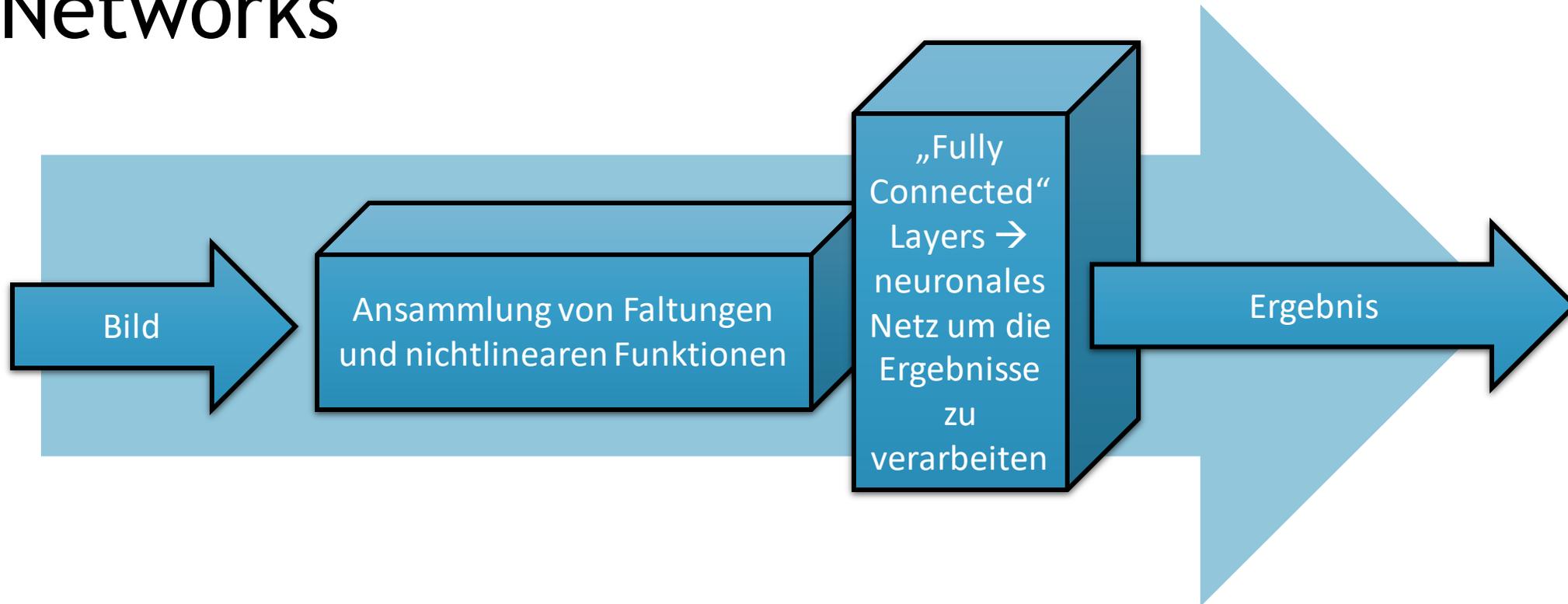
- + Komplexe Zusammenhänge von Bilddaten und erwartetem Ergebnis können nun trainiert werden
- + Kombinationen von Filtern und nichtlinearen Operationen
- + Erstellen der Filtermasken muss nicht mehr per Hand vorgenommen werden

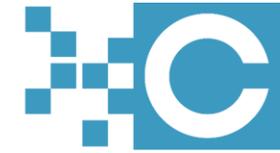
Nachteile:

- Es werden „gelabelte“ Daten benötigt
- Es werden VIELE Daten benötigt

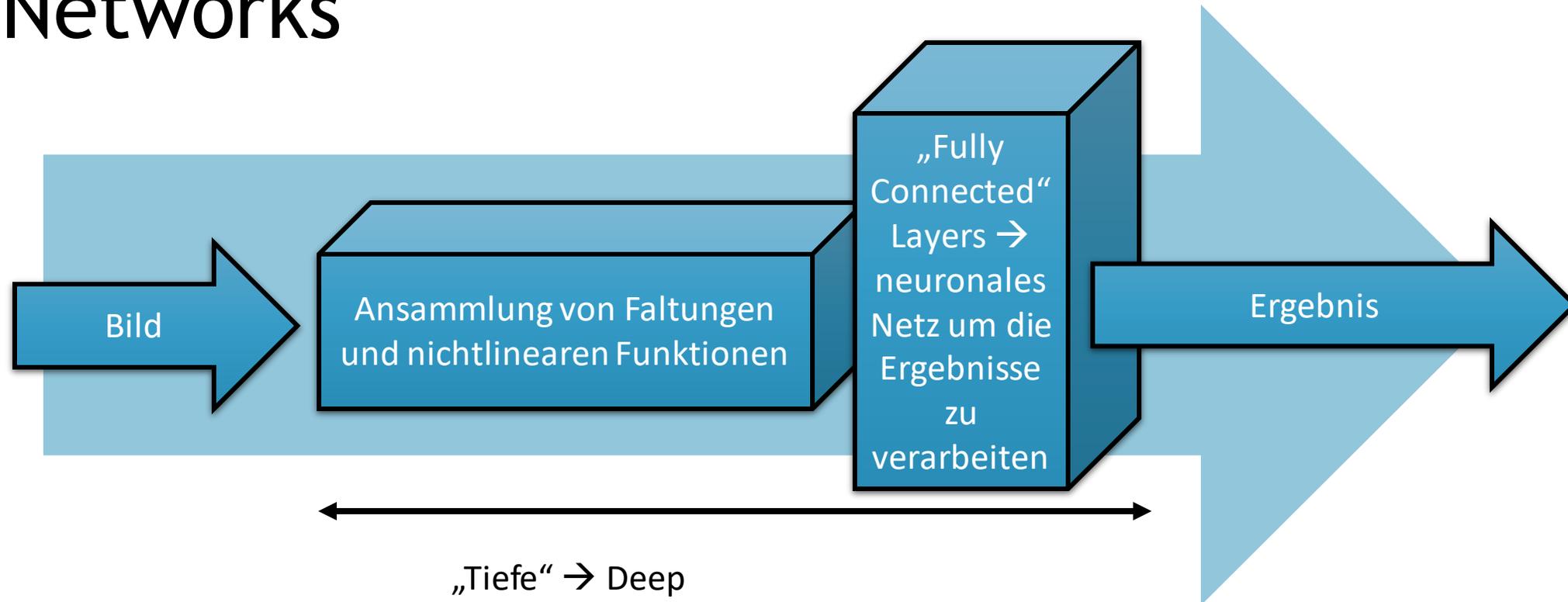


Deep Convolutional Neural Networks





Deep Convolutional Neural Networks





CIRCULAR INFINITY GMBH

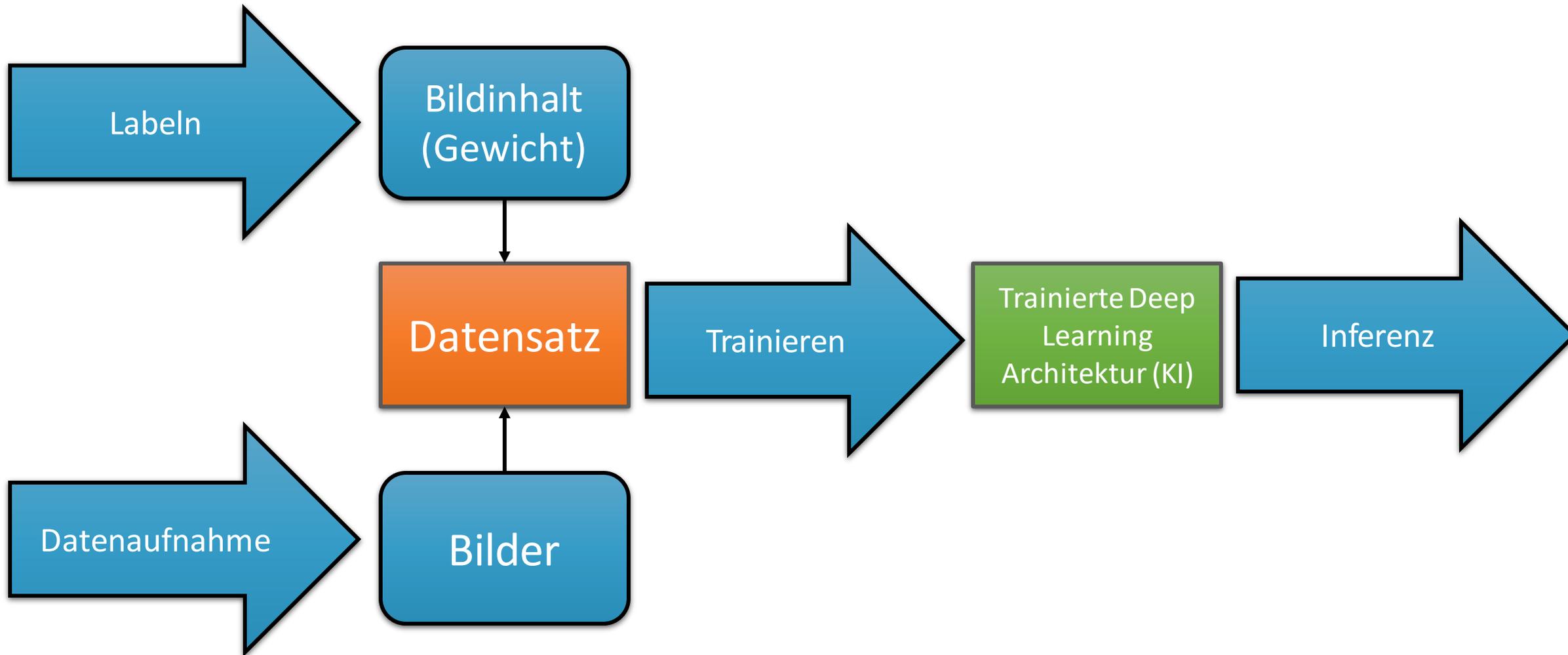
Methodik

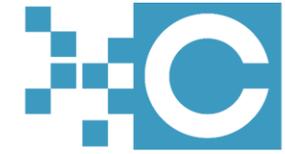
Einleitung | Grundlagen | **Methodik** | Ergebnisse | Fazit & Zukunft

Methodik



CIRCULAR INFINITY GMBH



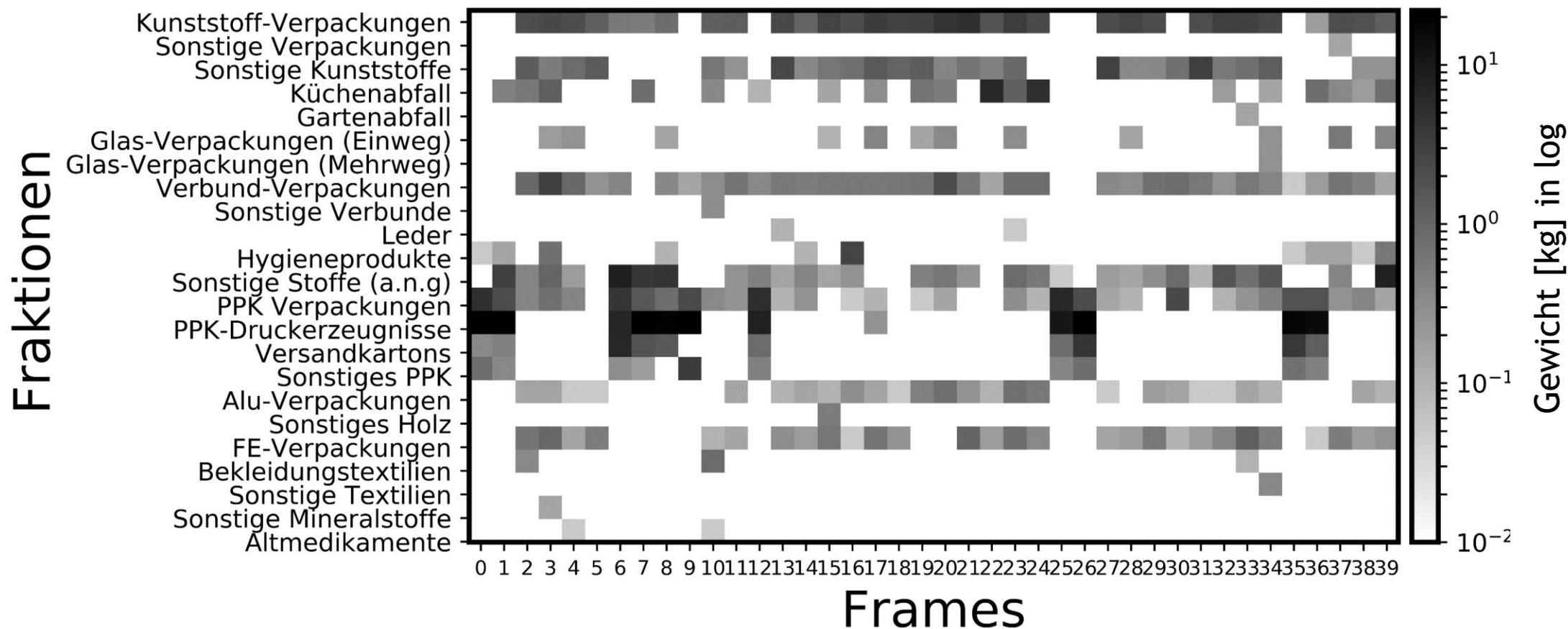


Labeln und Datenaufnahme

- Händische Sortierung und Messung des Gewichtes von realen, siedlungsabfallwirtschaftlichen Stoffströmen
- 2. Differenzierungsebene der Sächsischen Sortierrichtlinie
- Leichtverpackung (LVP) und Papier, Pappe und Kartonagen (PPK)
- Aufnahme mit einer Embedded-Hardware vor Ort und einer 720p Kamera
- 40 Bilder, 23 Kategorien, 496,34 kg



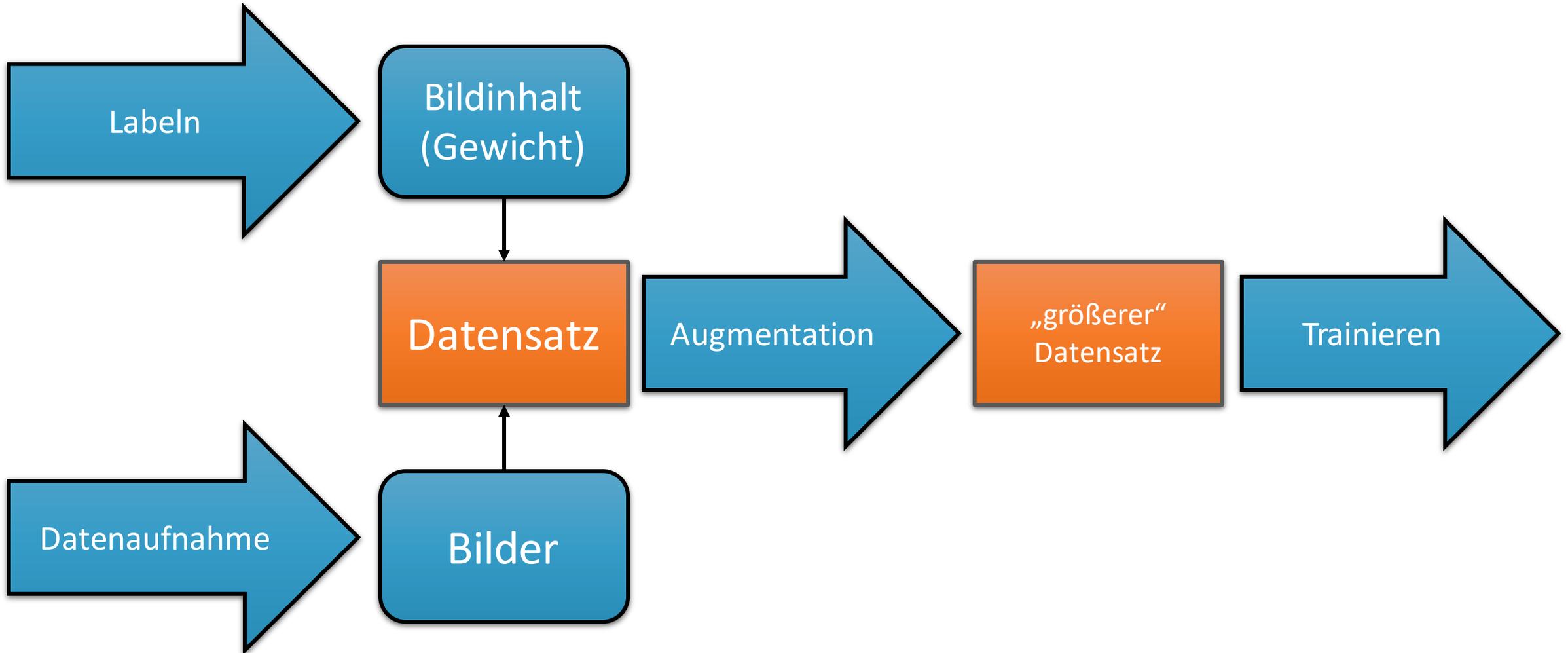
Gewichtsverteilung



Methodik

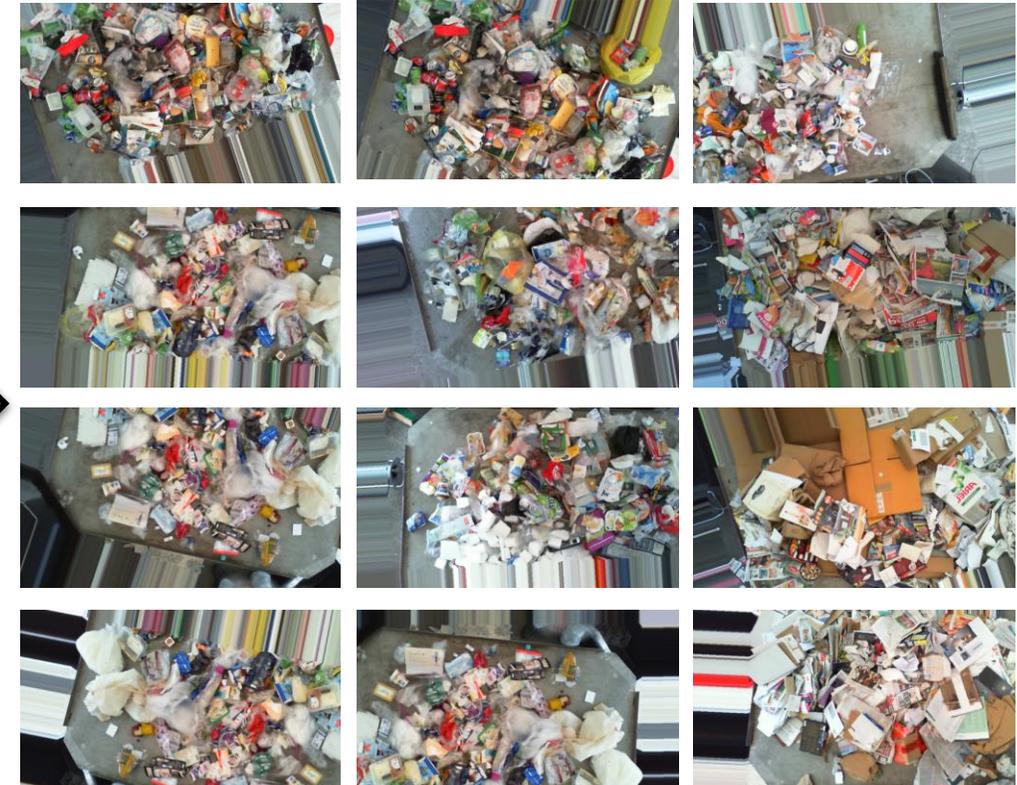


CIRCULAR INFINITY GMBH





Augmentationsmethoden

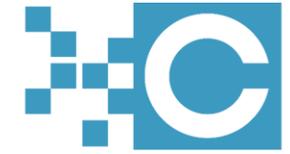




Trainieren

- Python Framework Keras [12] mit Tensorflow-Backend [11]
 - GPU (beschleunigt rechenintensive Faltungskerne durch Parallelisierung)
- Stochastic Gradient Descent Optimizer
- Early-Stopping
- Cyclic Learning Rate [5]

Architekturen



CIRCULAR INFINITY GMBH

- InceptionResNetV2 [6]
- InceptionV3 [7]
- MobileNet [8]
- ResNet50 [9]
- VGG16 [10]
- VGG19 [10]

Architekturen



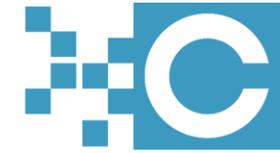
CIRCULAR INFINITY GMBH

- InceptionResNetV2 [6]
- InceptionV3 [7]
- MobileNet [8]
- ResNet50 [9]
- VGG16 [10]
- VGG19 [10]

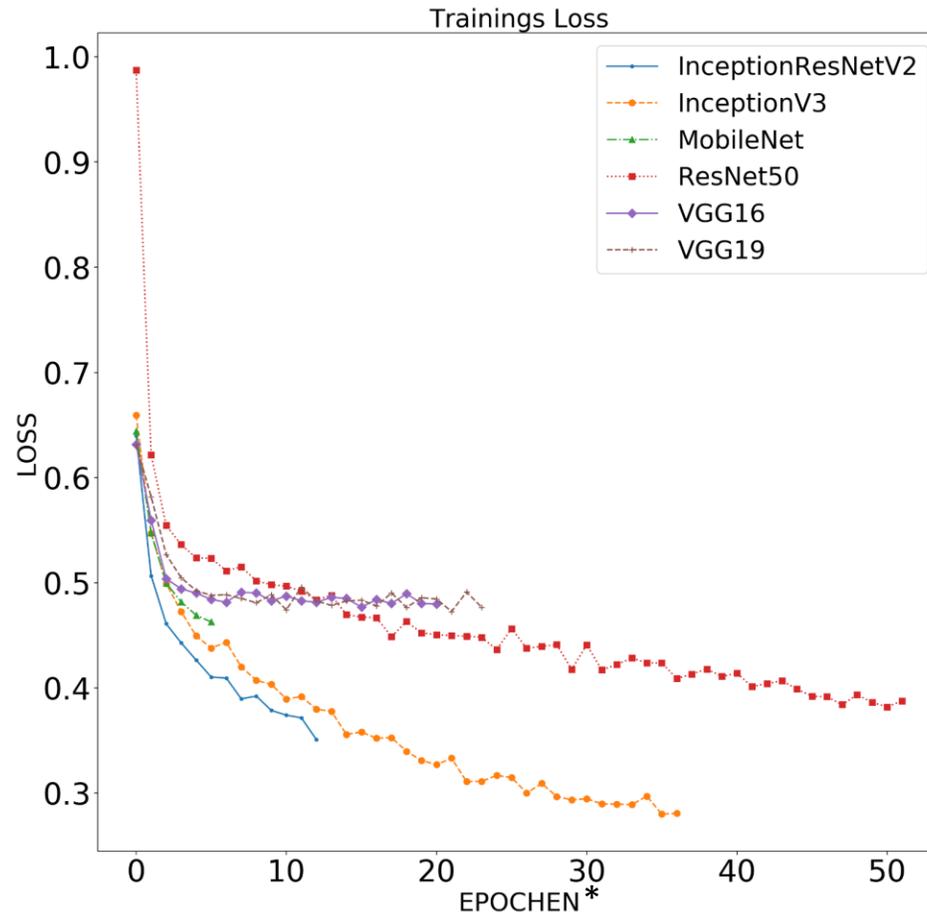


→ Letztes Fully Connected Layer mit einer linearen Aktivierungsfunktion

→ Regression statt Klassifikation



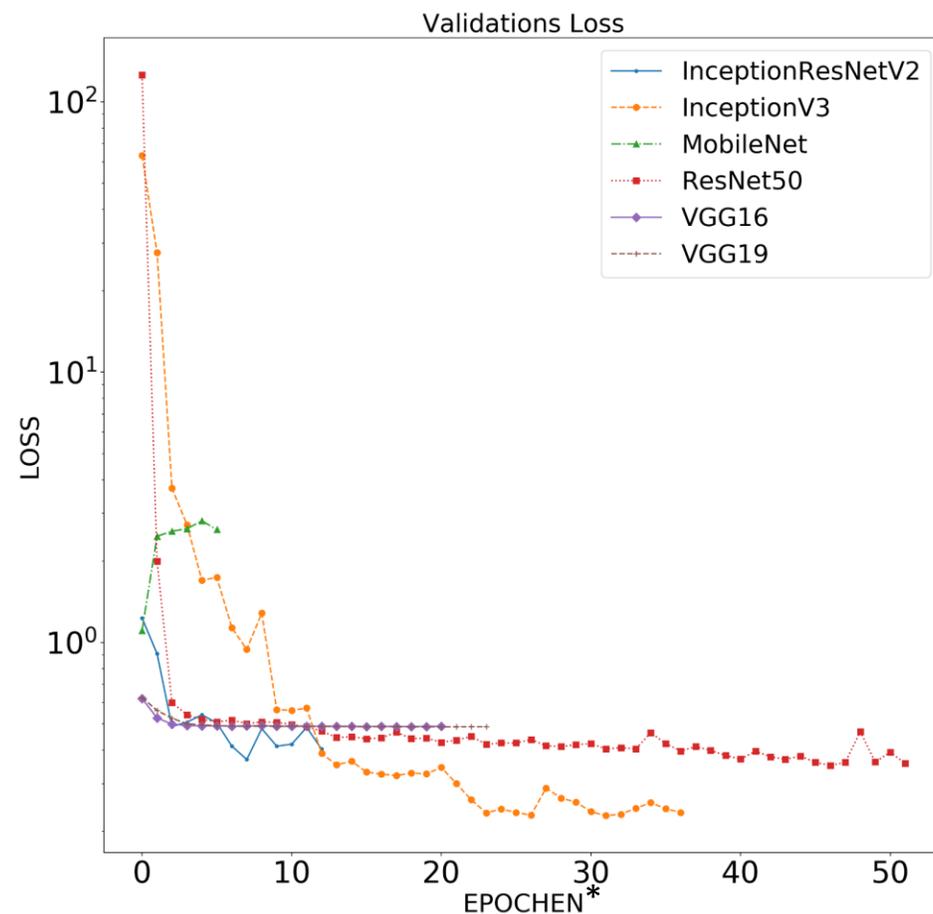
Trainings-Ergebnis



- 70% der Daten wurden für das Training verwendet
- Es wurde auf dem Z-score der Target-Vektoren trainiert und validiert
- Label-Fuzzyfication [4]

* Nach einer Epoche wurde einmal über den Datensatz iteriert

Validations-Ergebnis

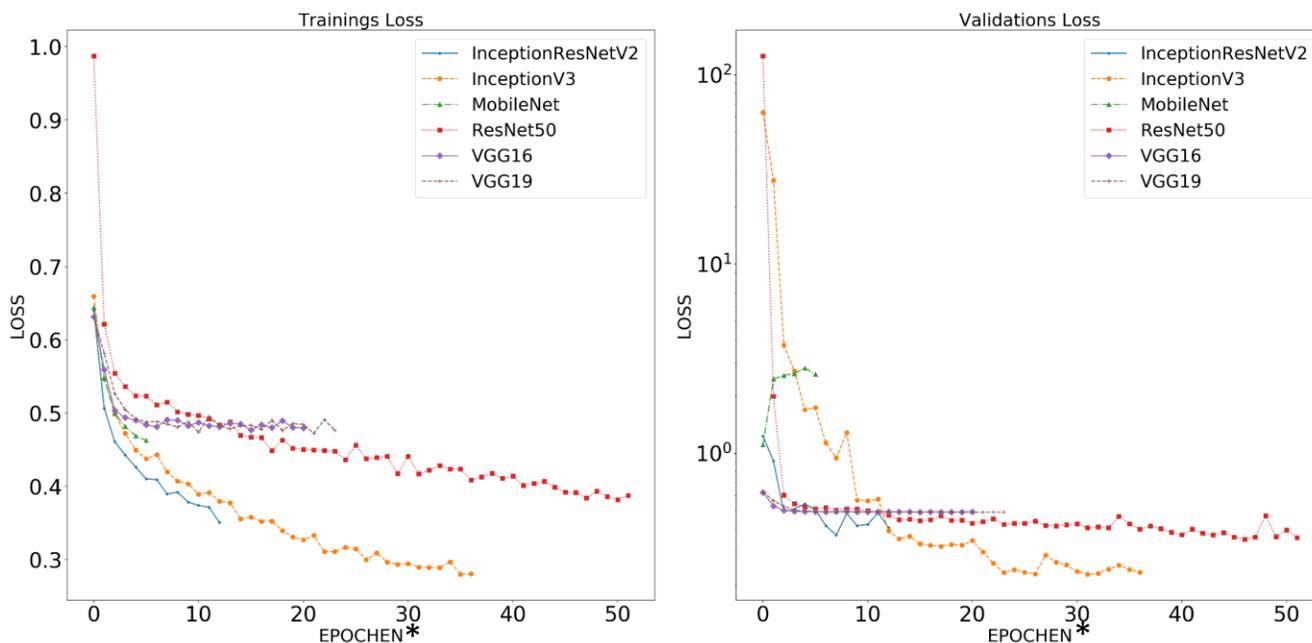


- 20% der Daten wurden zur Validierung (validation) verwendet
- Es wurde auf dem Z-score der Target-Vektoren trainiert und validiert
- Label-Fuzzyfication [4]

* Nach einer Epoche wurde einmal über den Datensatz iteriert



Trainings- und Validations-Ergebnis

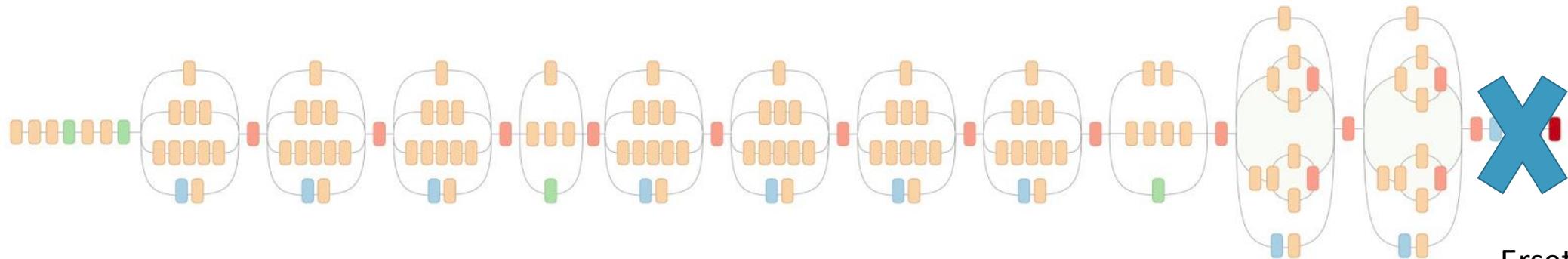


- Es wurde ein Maximal-Wert für die Stoffanteile basierend auf den Maximalwert in den Trainingsdaten pro Stoff eingeführt
- InceptionV3 Architektur schließt am besten ab
- MobileNet schließt am schlechtesten ab

Architektur	InceptionResNetV2	InceptionV3	MobileNet	ResNet50	VGG 16	VGG 19
Training	0.351	0.28	0.463	0.382	0.48	0.47
Validation	0.369	0.229	1.108	0.351	0.49	0.49

* Nach einer Epoche wurde einmal über den Datensatz iteriert

Inception V3 Architektur



- Convolution
- AvgPool
- MaxPool
- Concat
- Dropout
- Fully connected
- Softmax

Ersetzt mit
passenden
Layern (siehe
Folie 36)

Bildquelle: [13]



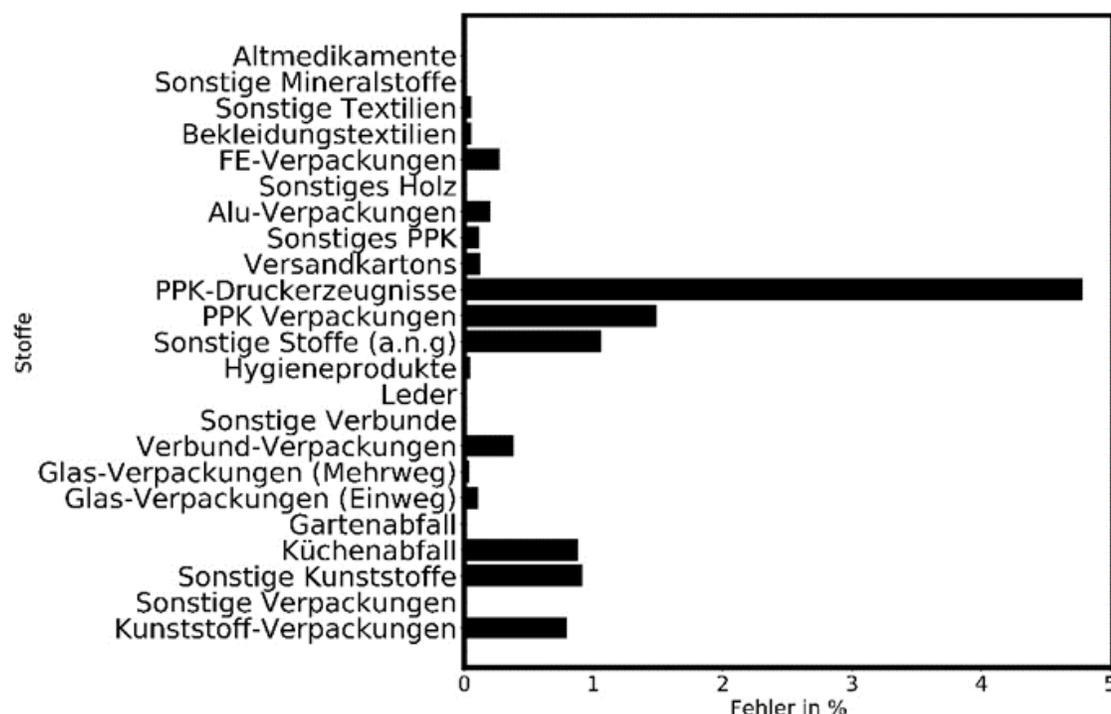
CIRCULAR INFINITY GMBH

Ergebnisse

Einleitung | Grundlagen | Methodik | Ergebnisse | Fazit & Zukunft

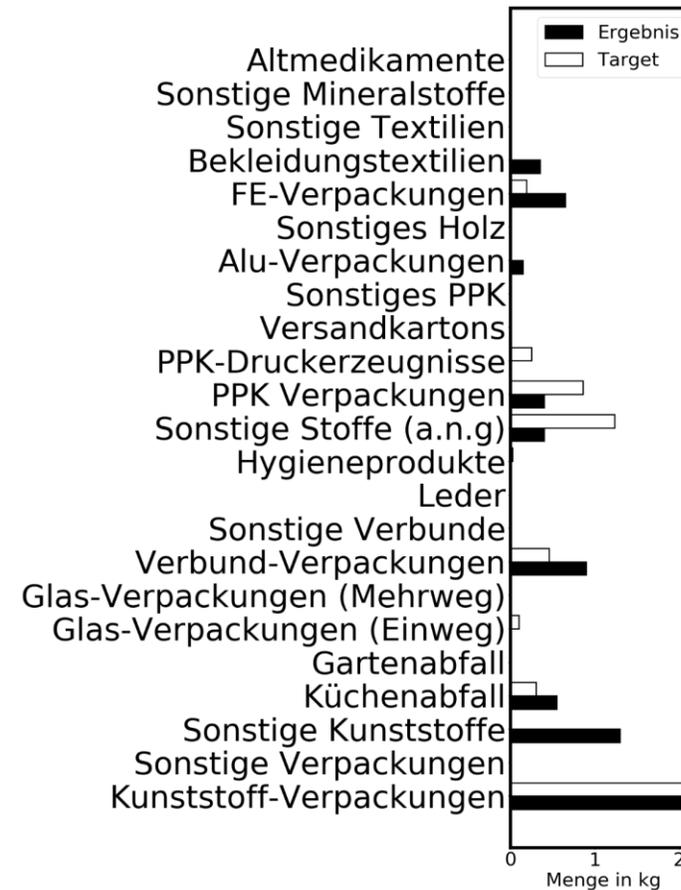
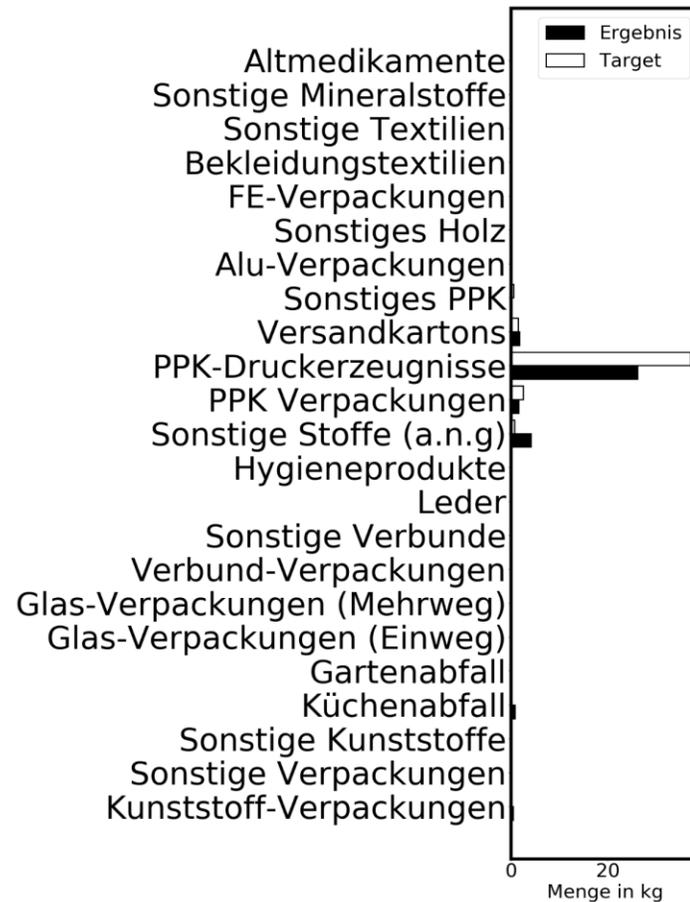


Ergebnis über Test-Daten



- Test auf 10% der originalen Daten
- Daten die noch nie zuvor gesehen wurden
- Fehler < 5%

Exemplarische Ergebnisse





CIRCULAR INFINITY GMBH

Fazit & Zukunft

Einleitung | Grundlagen | Methodik | Ergebnisse | **Fazit & Zukunft**



Fazit und Zukunft

- Es ist möglich anhand von Bilddaten mittels Deep Learning das Gewicht von Stoffen zu bestimmen, mit einer Genauigkeit von 99.5%
- Erweiterung des Datensatzes
- Untersuchung des erhöhten Fehlers bei PPK-Druckerzeugnissen
- Lösungsansatz auf andere Domänen und Fachbereiche übertragen



CIRCULAR INFINITY GMBH

Vielen Dank für Ihre Aufmerksamkeit
Für Fragen stehe ich Ihnen gerne jetzt, oder am Plakat zur
Verfügung

circular infinity GmbH

Künstliche Intelligenz in der Kreislaufwirtschaft

Neidenburger Straße 43, 45897 Gelsenkirchen

Telefon: + 49 (0) 151/23545905

Mail: info@circular-infinity.de

Internet: www.circular-infinity.de

Geschäftsführung: Tobias Althoff, Lutz Baberg

Quellen



CIRCULAR INFINITY GMBH

- [1] Scheelhaase, T.; Zinke, G.: Potenzialanalyse einer zirkulären Wertschöpfung im Land Nordrhein-Westfalen. Berlin, Hamburg, Düsseldorf 2016.
- [2] Müller, R.: Richtlinie zur einheitlichen Abfallanalytik in Sachsen. Sächsische Sortierrichtlinie 2014. Dresden 2015.
- [3] Keras Documentation: Image Preprocessing. ImageDataGenerator class. URL: <https://keras.io/preprocessing/image/>. Abrufdatum 25.09.2018.
- [4] Yang, C.-Y.; Chou, J.-J.; Lian, F.-L.: Robust classifier learning with fuzzy class labels for large-margin support vector machines. In: Neurocomputing 99 (2013), S. 1-14.
- [5] Smith, L.: Cyclical Learning Rates for Training Neural Networks. Washington 2017.
- [6] Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A.: Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning 2016.
- [7] Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z.: Rethinking the Inception Architecture for Computer Vision 2015.
- [8] Howard, A.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andretto, M.; Adam, H.: MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications 2017.
- [9] He, K.; Zhang, X.; Ren, S.; Sun, J.: Deep Residual Learning for Image Recognition 2015.
- [10] Simonyan, Karen; Zisserman, Andrew: Very deep convolutional networks for large-scale image recognition 2015.
- [11] Martín Abadi; Ashish Agarwal; Paul Barham; Eugene Brevdo; Zhifeng Chen; Craig Citro; Corrado, G. S.; Andy Davis; Jeffrey Dean; Matthieu Devin; Sanjay Ghemawat; Ian Goodfellow; Andrew Harp; Geoffrey Irving; Michael Isard; Jia, Y.; Rafal Jozefowicz; Lukasz Kaiser; Manjunath Kudlur; Josh Levenberg; Dandelion Mané; Rajat Monga; Sherry Moore; Derek Murray; Chris Olah; Mike Schuster; Jonathon Shlens; Benoit Steiner; Ilya Sutskever; Kunal Talwar; Paul Tucker; Vincent Vanhoucke; Vijay Vasudevan; Fernanda Viégas; Oriol Vinyals; Pete Warden; Martin Wattenberg; Martin Wicke; Yuan Yu; Xiaoqiang Zheng: TensorFlow. Large-Scale Machine Learning on Heterogeneous Systems (2015). URL: <https://www.tensorflow.org/>.
- [12] Chollet, F.; others: Keras (2015).
- [13] <https://ai.googleblog.com/2016/08/improving-inception-and-image.html> Aufgerufen am 03.11.2018
- [14] Toghi, Behrad; Grover, Divas: MNIST Dataset Classification Utilizing k-NN Classifier with Modified Sliding Window Metric