

Abschlussvortrag Bachelorarbeit

Hauke Hinrichs

Lokalisierung von Windenergieanlagen aus Satellitenbildern mittels Convolutional Neural Networks

25.09.2020

LEIBNIZ UNIVERSITÄT HANNOVER, INSTITUT FÜR INFORMATIONSVERRARBEITUNG



Agenda

1. Motivation
2. Lösungsansatz
3. Daten
4. Experimente
5. Ergebnisse
6. Zusammenfassung und Ausblick

1. Motivation

Standorte von WEA essenziell für diverse Forschungsbelange

Qualitativ gute öffentliche
Datenbanken sind rar

Pflege einer WEA Datenbank
aufwändig

Ziel 1: Überprüfung
von (alten) Standorten

Ziel 2: Lokalisierung von
neuen Standorten



2. Lösungsansatz – Objekt Detektierung

Computer Vision: **Objekt Detektierung** mittels Convolutional Neural Networks

Im Vergleich zur **Klassifikation**: Variable Ausgabedimension und Lokalisierung von Objekten im Bild mittels Bounding Boxen

Voraussetzung: Trainingsdaten annotiert mit Bounding Boxen

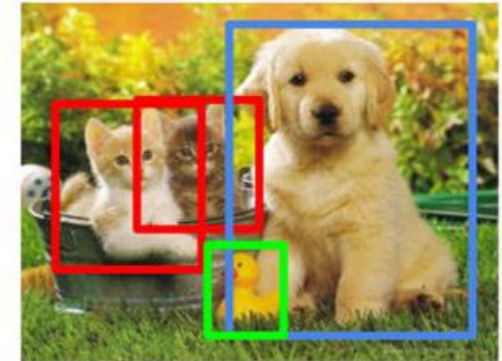
Modelle: SSD, YOLOv4, Faster R-CNN

Classification



CAT

Object Detection



CAT, DOG, DUCK

<https://www.datacamp.com/community/tutorials/object-detection-guide>

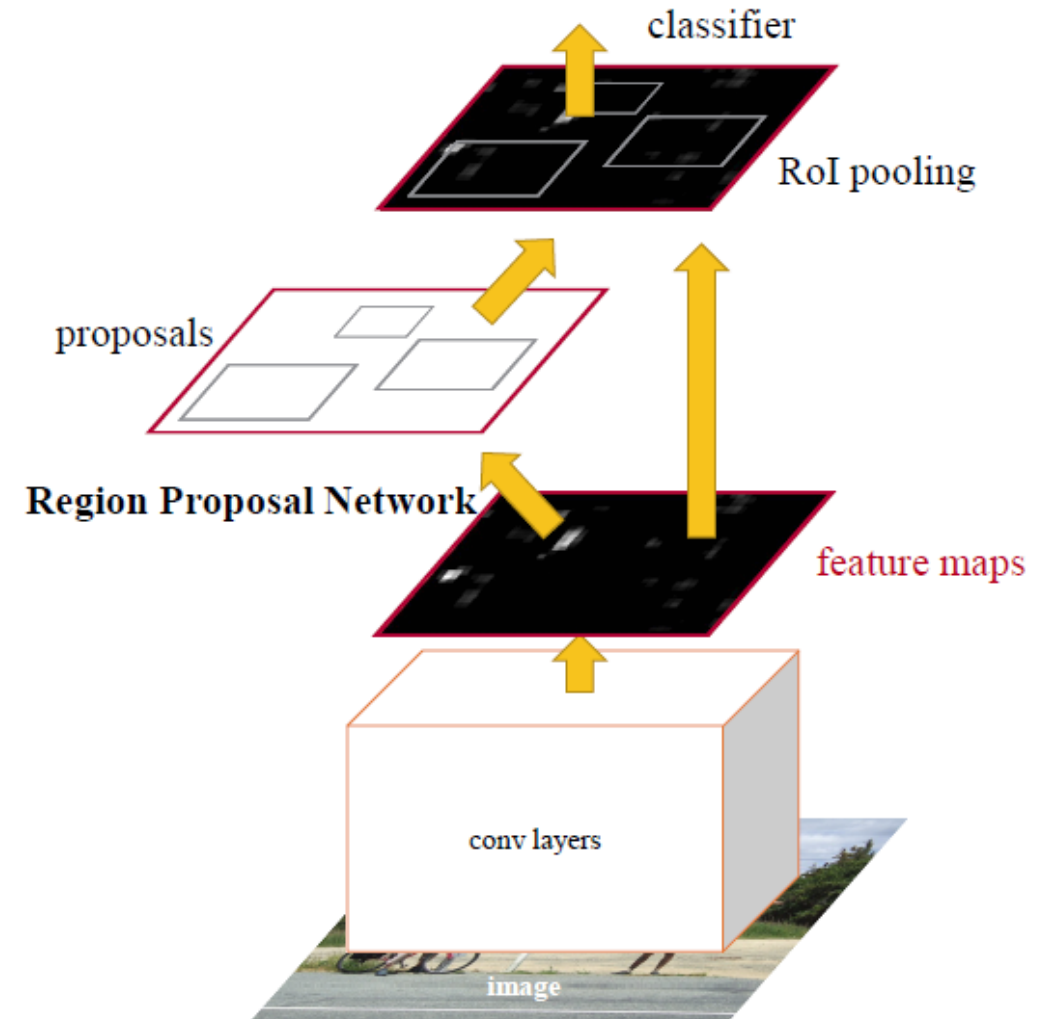
2. Lösungsansatz – Faster R-CNN

Fully Convolutional Net erzeugt Feature Maps

Region Proposal Network produziert aus Feature Maps Objektvorschläge (RoI)

RoI Pooling bringt RoI auf feste Größe (bspw. 7x7)

Classifier klassifiziert jede RoI



S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in IEEE Transactions on Pattern Analysis and Machine Intelligence

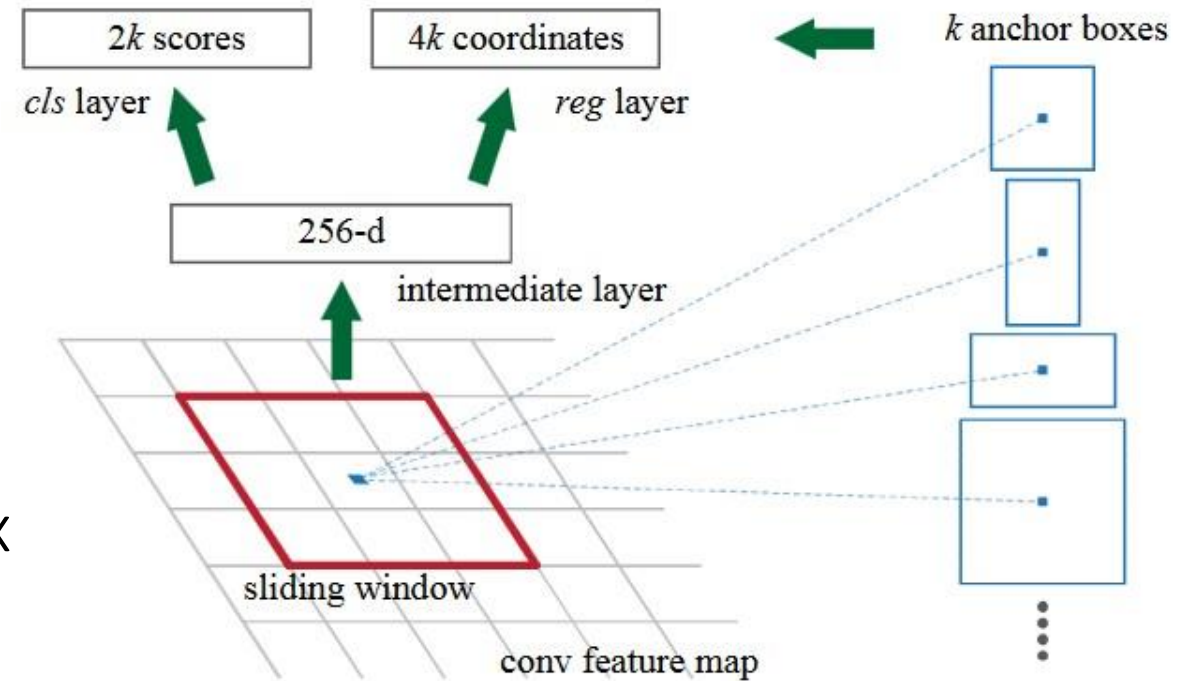
2. Lösungsansatz – RPN

Jede Position (**Anchor**) des Sliding Window produziert k **Anchor Boxes**

Anchor Box Größen * Anchor Box
Seitenverhältnisse = k

Output:

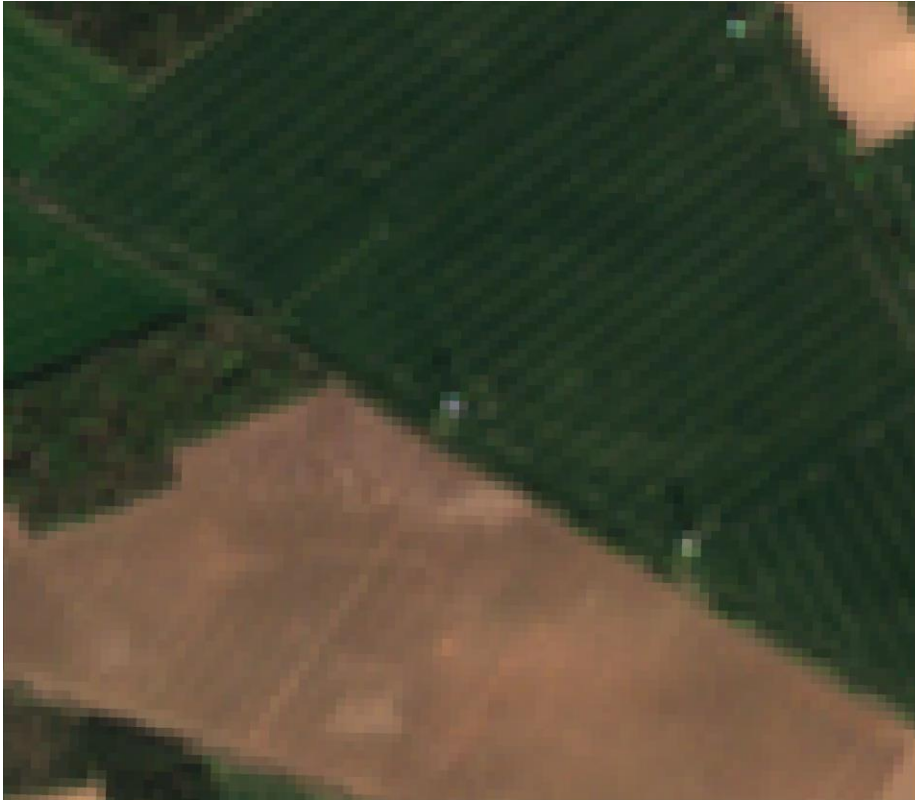
- cls layer: Objekt Score
- reg layer: Anpassung der Anchor Box



S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in IEEE Transactions on Pattern Analysis and Machine Intelligence

3. Daten – Auswahl der Bildquelle

ESA Sentinel-2



- 10x10m pro Pixel
- Wolken

Google Maps Static API



- + ca. 0,7x0,7m pro Pixel
- unvollständige Meta-Informationen

Imagery ©2020 , GeoBasis-DE/BKG, GeoContent, Maxar Technologies

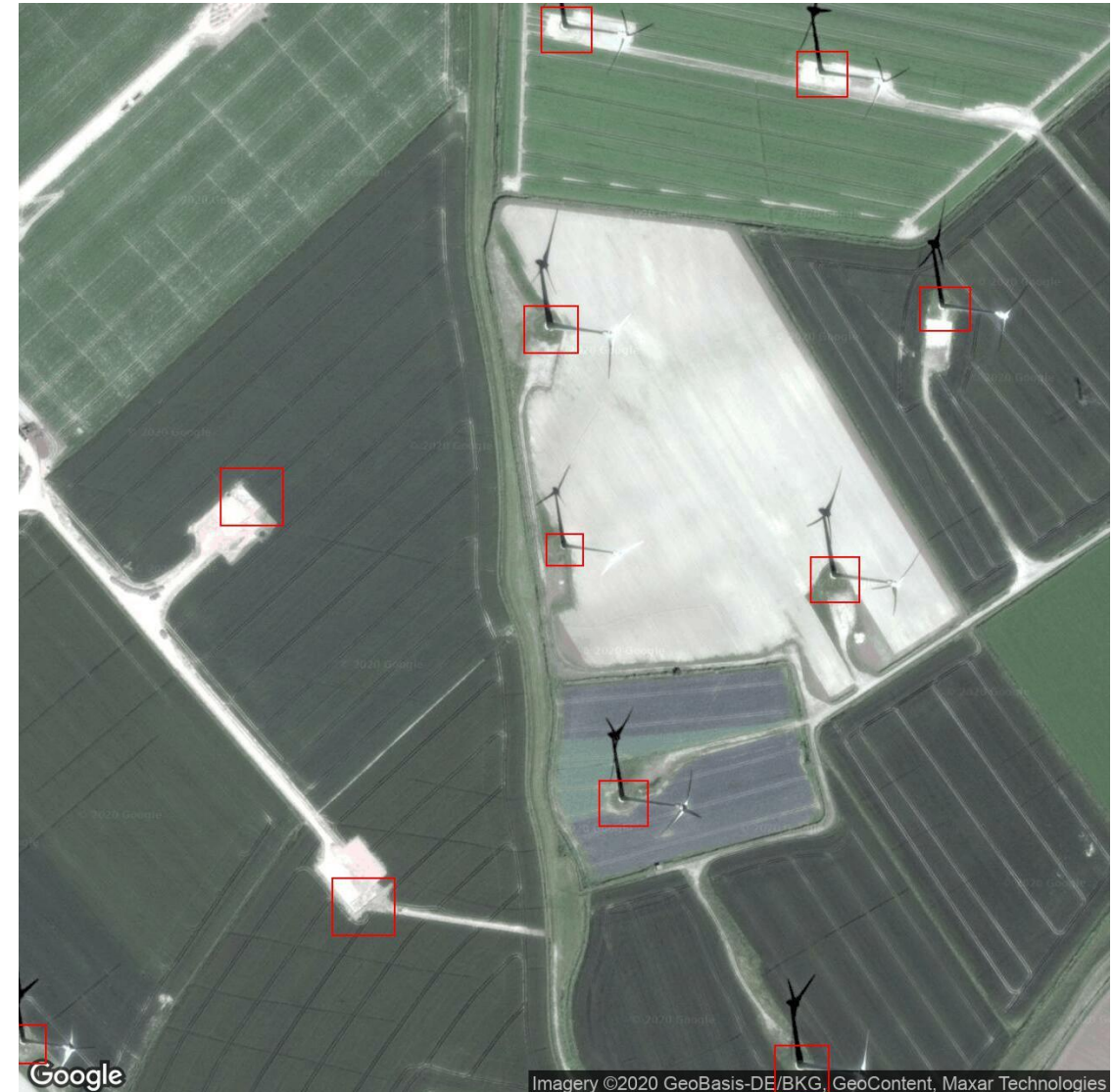
3. Daten – Annotation von Boxen

Aktuelle Datenbank aller WEA
Deutschlands, sowie den USA

Autom. Annotation der Bounding Boxes
in Abhängigkeit der WEA Größe

Problem: Datum der Aufnahme nicht
bekannt -> 2 Arten Fehlannotationen

1. Box ohne Anlage
2. Anlage ohne Box



3. Daten – Evaluation der Datenqualität

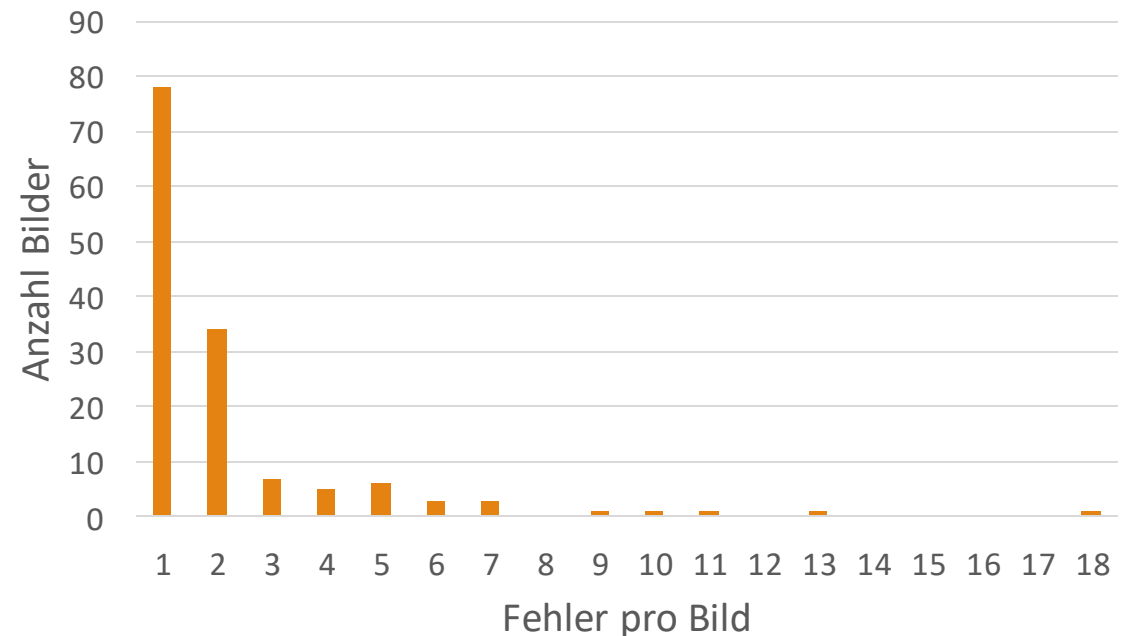
Sample mit 2000 Bildern:

104 Boxen ohne Anlage, 209 Anlagen ohne Box

141 (7%) Bilder enthielten Fehler

Im Mittel 4,2 Boxen pro Bild

=> Von den ca. 8400 Boxen
im Sample sind 1,2% falsch

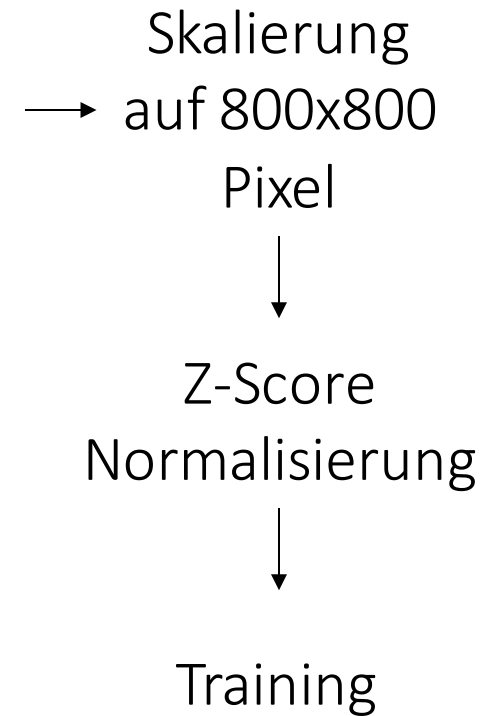


3. Daten – Augmentation & Normalisierung

Rotation, vert. und hor. Flips, Entfernung des Labels, Translation

1280x1280 Pixel

1030x1030 Pixel



3. Daten – Set Splits

Geschachtelte Umkreissuche \Rightarrow Ein Windpark taucht nur in einem Set auf

Set Größen:

	Training	Validation	Test
DEU	17820	2970	2970
USA	5144	858	857

USA: Häufiger ältere Aufnahmen,
dadurch viel mehr Fehler



Händische Prüfung von 12000 Bildern:
6859 fehlerfrei



3. Daten – Unterschiede USA \leftrightarrow DEU

Struktur der Windparks unterscheidet sich z.T. stark von DEU

Häufiger sehr große Windparks an Standorten mit hohen Windaufkommen
WEA stehen dort meist sehr nah beieinander

Problem:

Überrepräsentation kleiner Anlagen

⇒ Nur Anlagen nach 2000 berücksichtigt



4. Experimente – Implementierung

Implementierung von Github:

<https://github.com/kentaroy47/frcnn-from-scratch-with-keras>

Keras 2.3.1 mit Tensorflow 2.1.0 Backend

Backbone Netz: **ResNet-50**

Vortrainierte Gewichte:

<https://github.com/fchollet/deep-learning-models/releases>

4. Experimente


GTB sind stets quadratisch und nehmen relativ wenig Platz im Bild ein \Rightarrow Anchor Box Seitenverhältnis (1:1)
Anchor Box Größen klein wählen

1. RPN allein
2. RPN + Classifier
3. RPN + Classifier größere GTB und Anchor Boxen
4. Geographische Abhängigkeit des Modells aus 3. am Beispiel der USA

4. Experimente – Post Processing

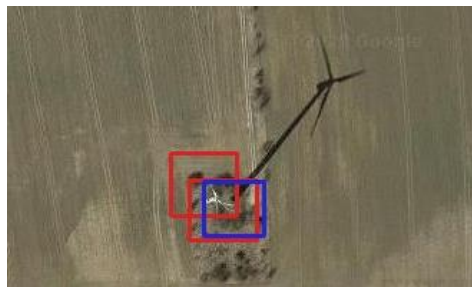
Prädizierte Boxen überlappen sich häufig
Lösung: **Non-Maximum-Suppression** (NMS)

Experimente 1 – 3: Strikte NMS
Schwellwert nur minimal größer als 0

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$


<https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/>

Ohne NMS:



Mit NMS:



4. Experimente – Metriken

TP: $\text{IoU}(\text{Prädizierte Box, GTB}) \geq \lambda_{IoU}$

FP: $\text{IoU}(\text{Prädizierte Box, GTB}) < \lambda_{IoU}$

FN: Es ex. für eine GTB keine Prädiktion mit $\text{IoU} \geq \lambda_{IoU}$

Precision (P), Recall (R), F1-Score ($F1$)

\bar{D} : Durchschnittliche Distanz der Mittelpunkte von Prädizierter und GTB in Metern von TPs

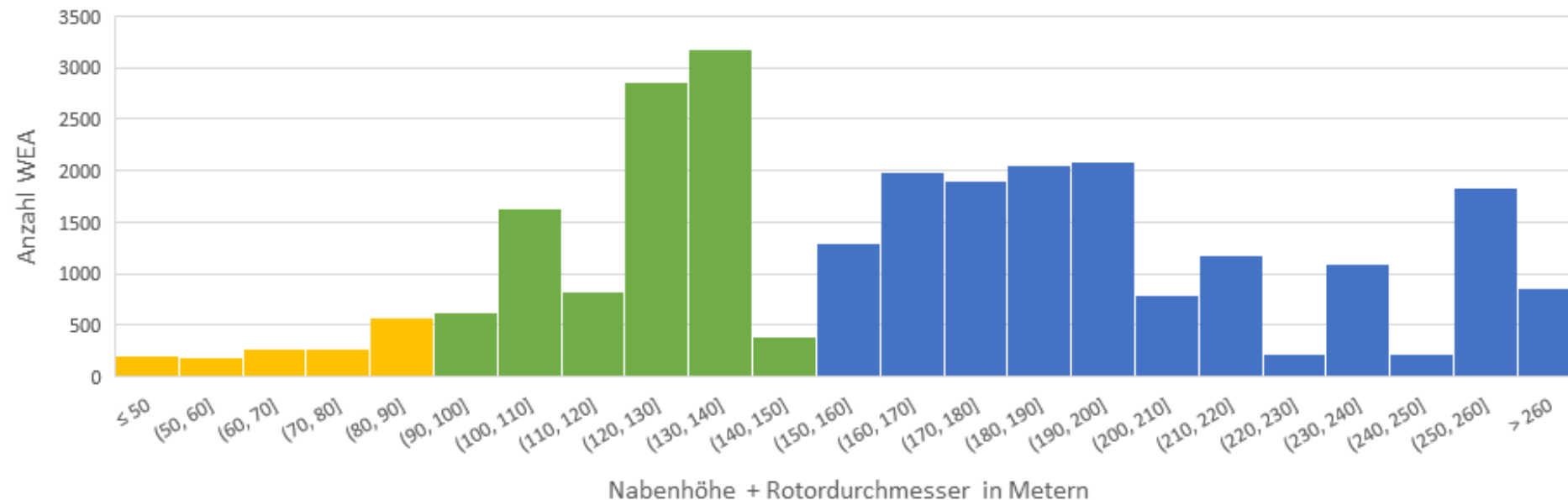
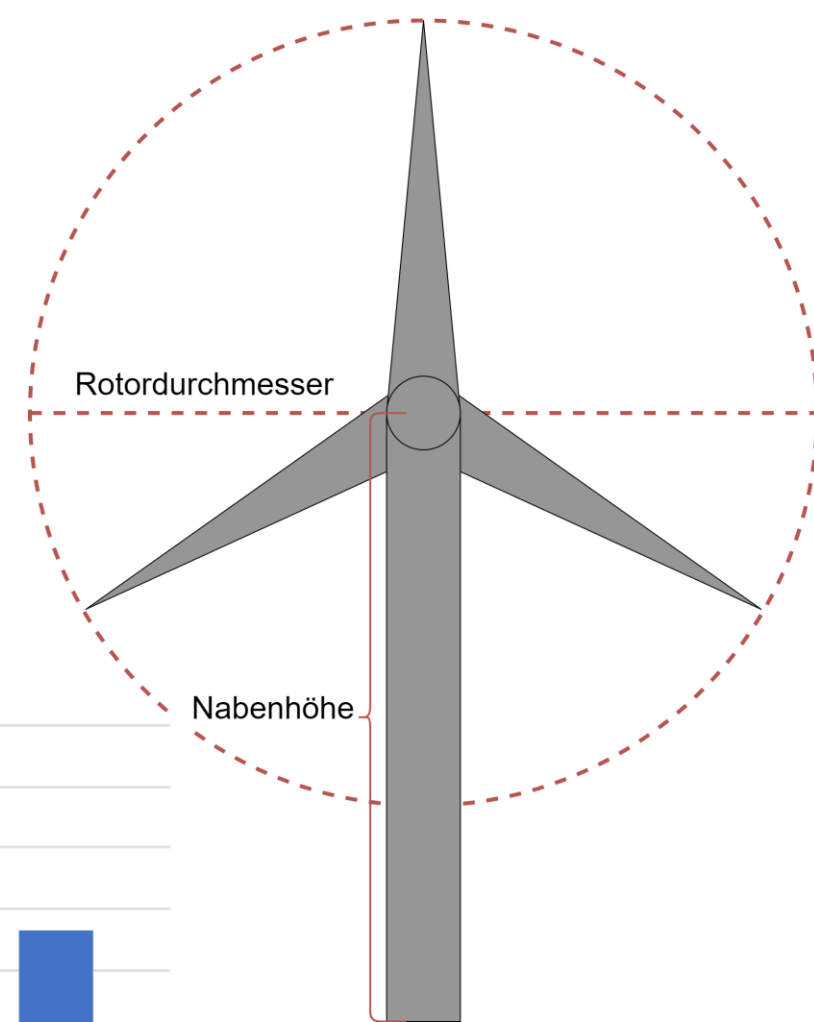
Variierung von λ_{IoU} um bestmögliche Balance zu finden

4. Experimente – Metriken

Anlagen werden in 3 Größenkategorien eingeteilt:

Klein, mittel und groß

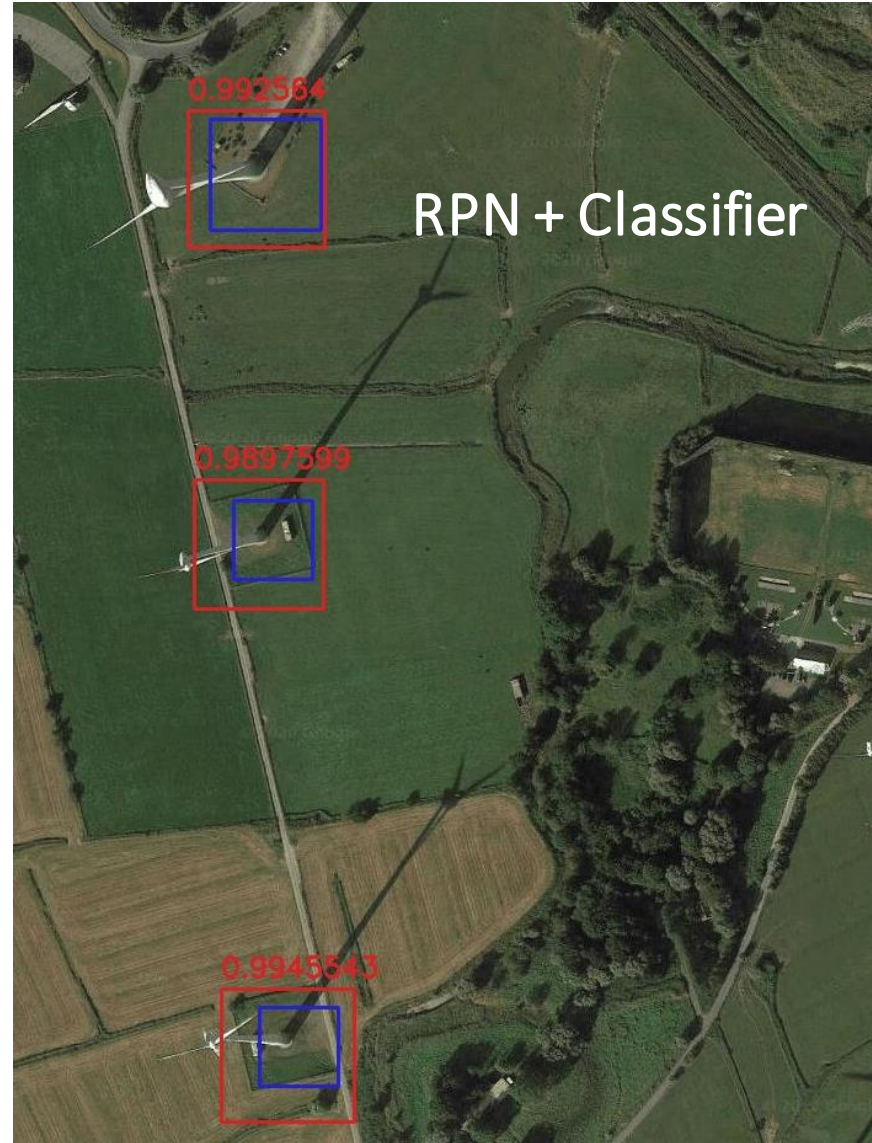
Werte werden für jede Kategorie separat ermittelt



5. Ergebnisse – RPN + Classifier

λ_{IoU}	P	R	$F1$	\bar{D}	R_k	R_m	R_g
0,1	0,98	0,74	0,84	7,87m	0,06	0,61	0,82
0,4	0,6	0,45	0,52	5,26m	0	0,1	0,68

5. Ergebnisse – RPN + Classifier



5. Ergebnisse – RPN + Classifier größere GTB

λ_{IoU}	P	R	$F1$	\bar{D}	R_k	R_m	R_g	
0,1	0,98	0,74	0,84	7,87m	0,06	0,61	0,82	Kleinere GTB
0,4	0,6	0,45	0,52	5,26m	0	0,1	0,68	
0,1	0,97	0,86	0,91	13,36m	0,6	0,84	0,89	Größere GTB
0,4	0,72	0,64	0,66	9,16m	0,003	0,57	0,73	

5. Ergebnisse – RPN + Classifier größere GTB

Kleinere GTB



Größere GTB

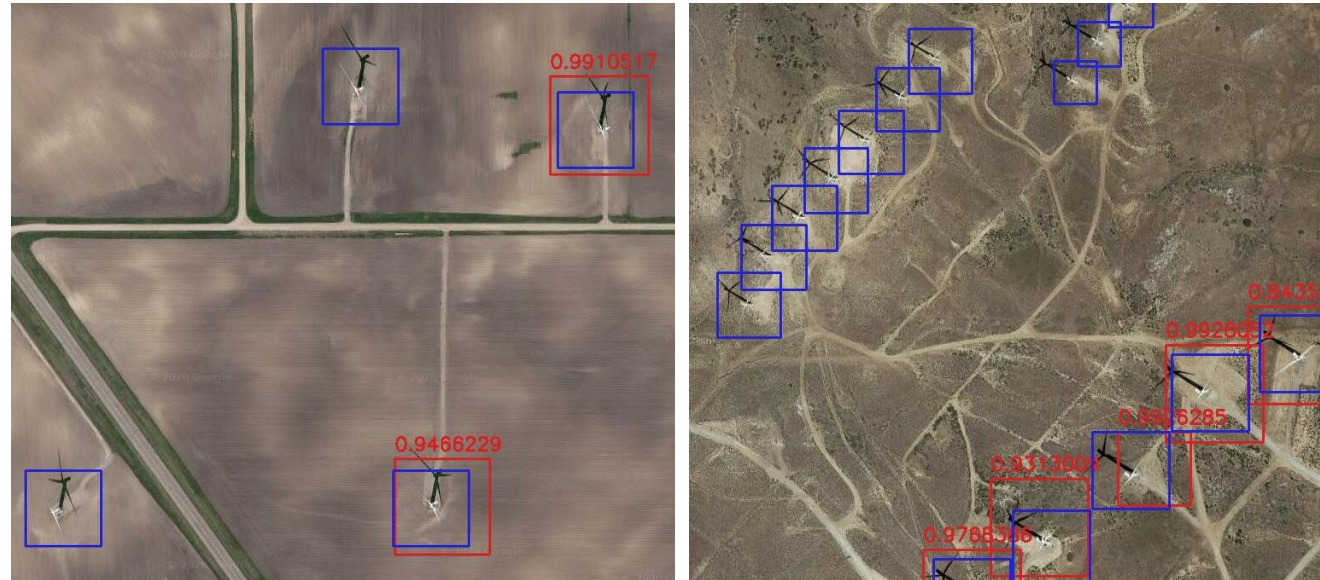


5. Ergebnisse – Geographische Abhängigkeit

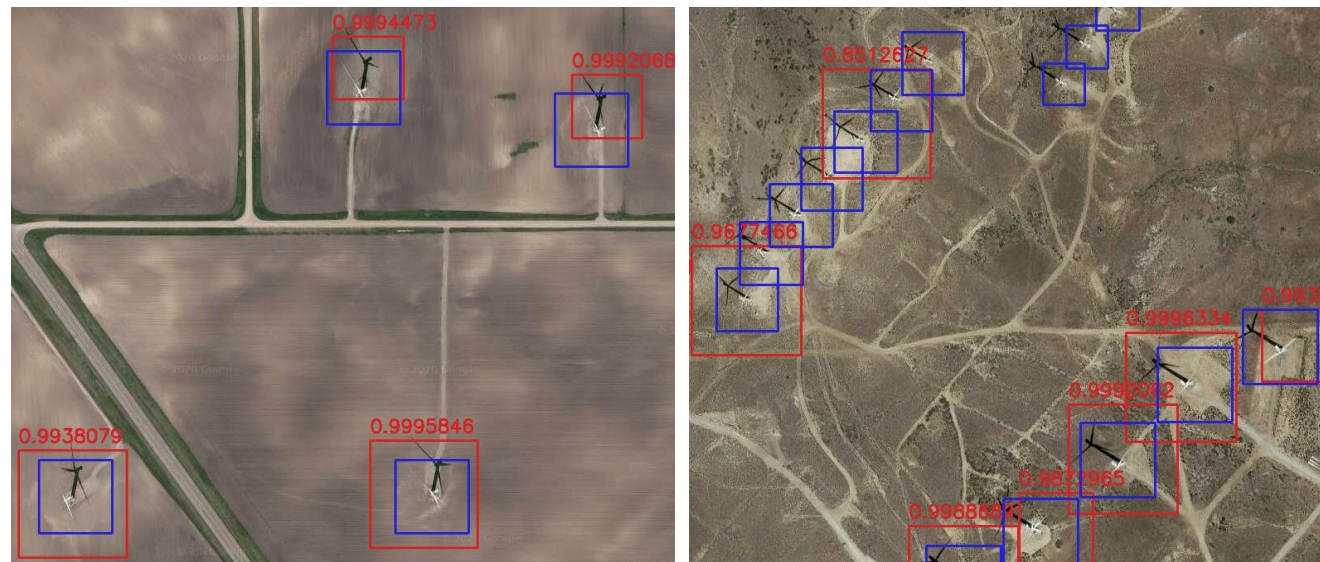
λ_{IoU}	P	R	$F1$	\bar{D}	R_k	R_m	R_g	
0,1	0,97	0,86	0,91	13,36m	0,6	0,84	0,89	DEU
0,4	0,72	0,64	0,66	9,16m	0,003	0,57	0,73	
0,1	0,88	0,72	0,79	15,34m	0,09	0,31	0,79	USA ohne Nachtraining
0,4	0,77	0,63	0,69	11,56m	0	0,2	0,7	
0,1	0,85	0,88	0,86	19,42m	0,17	0,75	0,93	USA mit Nachtraining
0,4	0,7	0,72	0,71	15,64m	0	0,27	0,82	

5. Ergebnisse – Geographische Abhängigkeit

Ohne Nachtraining



Mit Nachtraining



6. Zusammenfassung und Ausblick

Motivation: Pflege einer WEA Datenbank → Lösungsansatz: Objekt Detektierung

Verwendetes Modell: Faster R-CNN

Bilddaten über Google Static Maps API

Autom. Annotation von Bounding Boxen mithilfe vorliegender WEA Datenbank

Gute Ergebnisse für DEU

Modell	P	R	$F1$	\bar{D}
Kleinere GTB	0,98	0,74	0,84	7,87m
Größere GTB	0,97	0,86	0,91	13,36m

Möglichkeiten der Weiterentwicklung:

Domänenadaption (Problem mit Strukturunterschieden der Windparks)

Anpassung Verlust Funktion (RPN) an Problem, weniger Gewichtung auf IoU

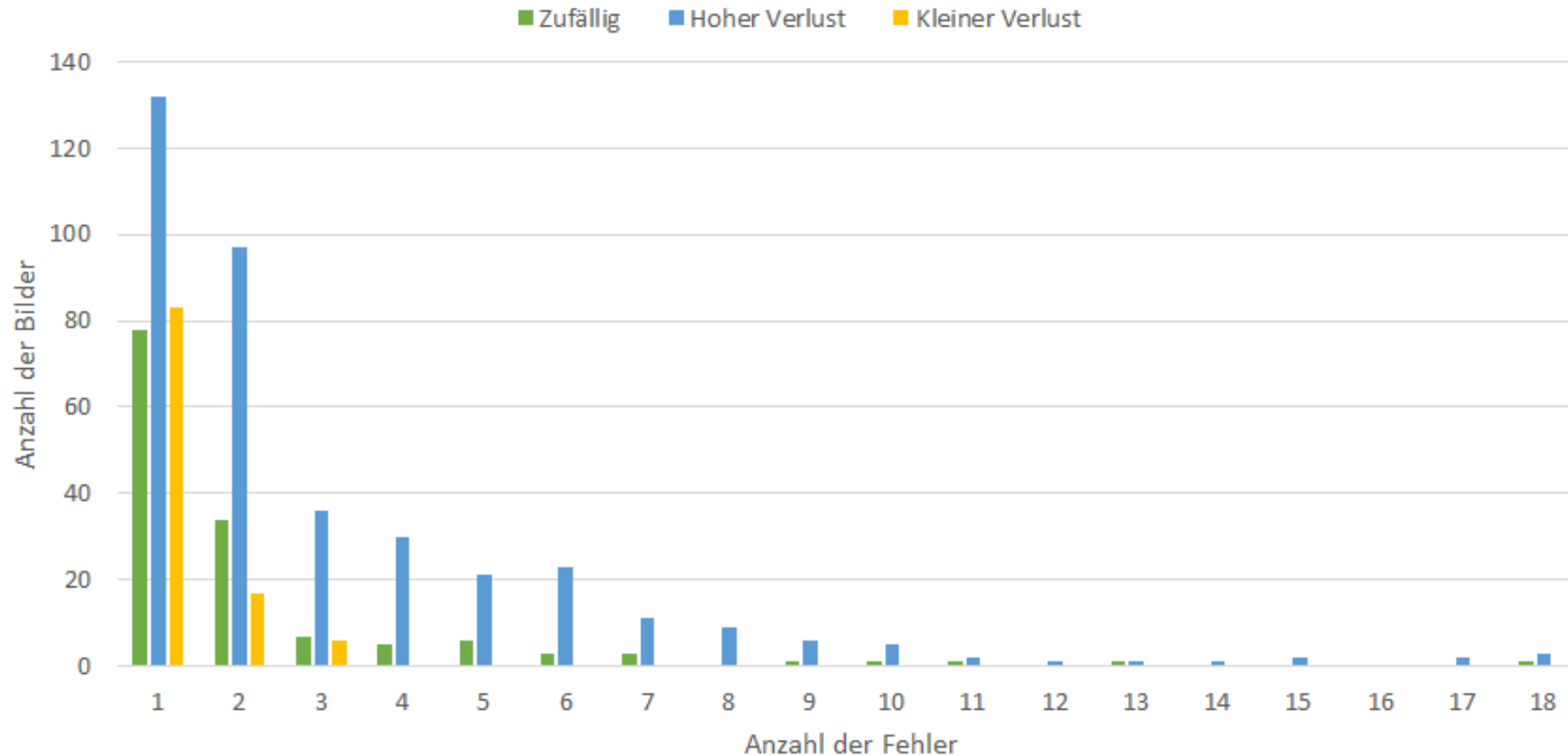
Experimente mit kompakteren Ansätzen

Backup Folien

Ergebnisse – RPN allein

λ_{obj}	λ_{IoU}	P	R	$F1$	\bar{D}	R_k	R_m	R_g
0,9	0,1	0,36	0,95	0,52	6,63	0,81	0,95	0,96
0,9	0,4	0,27	0,71	0,39	5,47	0,003	0,49	0,9
0,99999	0,1	0,6	0,9	0,72	6,45	0,68	0,87	0,93
0,99999	0,4	0,46	0,68	0,55	5,45	0,003	0,45	0,88

Daten – Evaluation der Datenqualität



Experiment – Modifiziertes RPN

Modifikation: Nur Koordinaten statt Bounding Boxen prädizieren

Training konvergiert nicht

