Erkennung von hand-geschriebenen Wörtern

KILIAN AARON BRINKNER

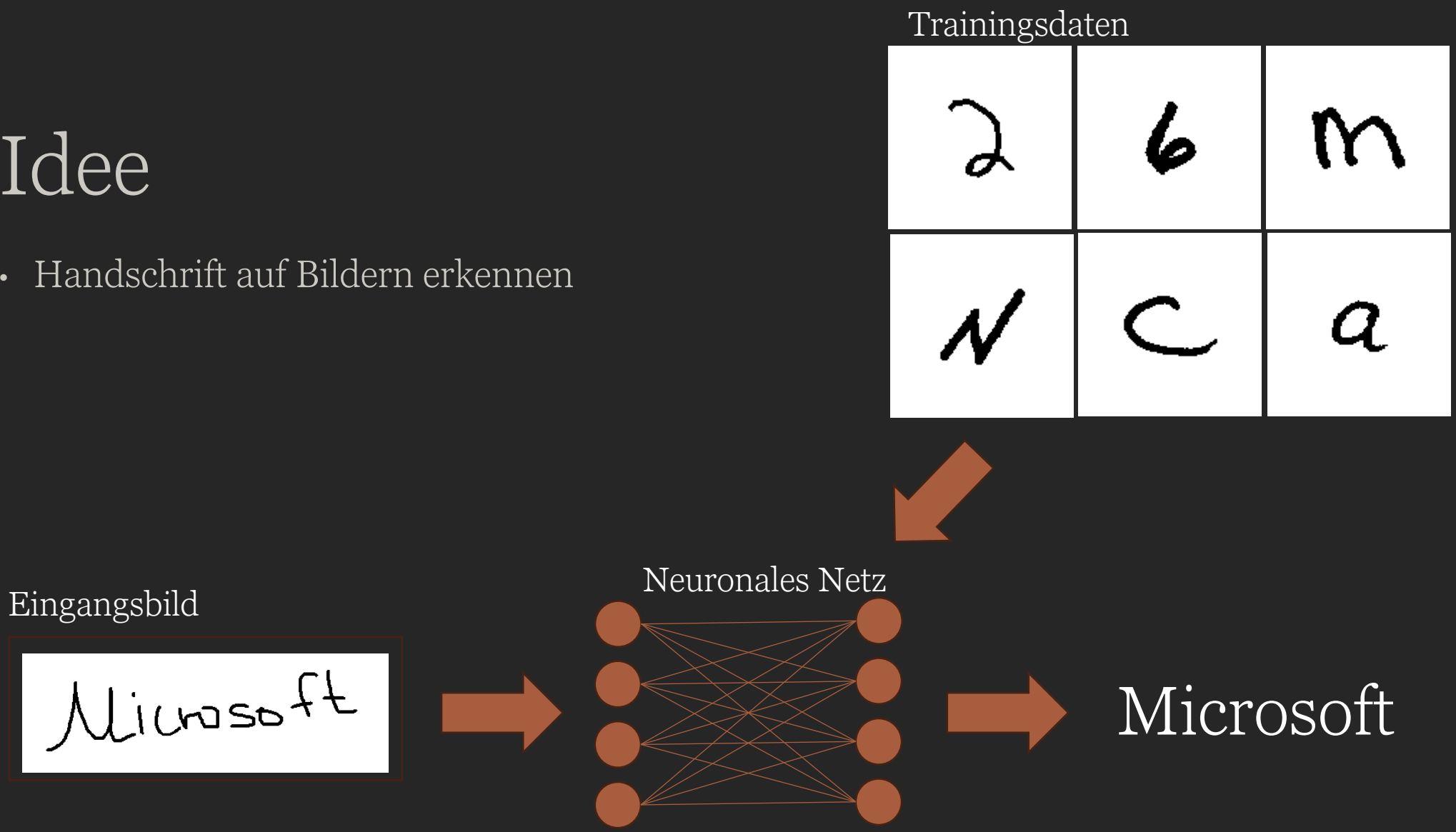
MASCHINELLES SEHEN

Gliederung

- Idee
- Vorgehen
- Datenaufbereitung
- Ergebnisse Buchstaben
- Ergebnisse Wörter
- Vergleich
- Fazit

Idee

- Handschrift auf Bildern erkennen



Vorgehen: Datenbeschaffung

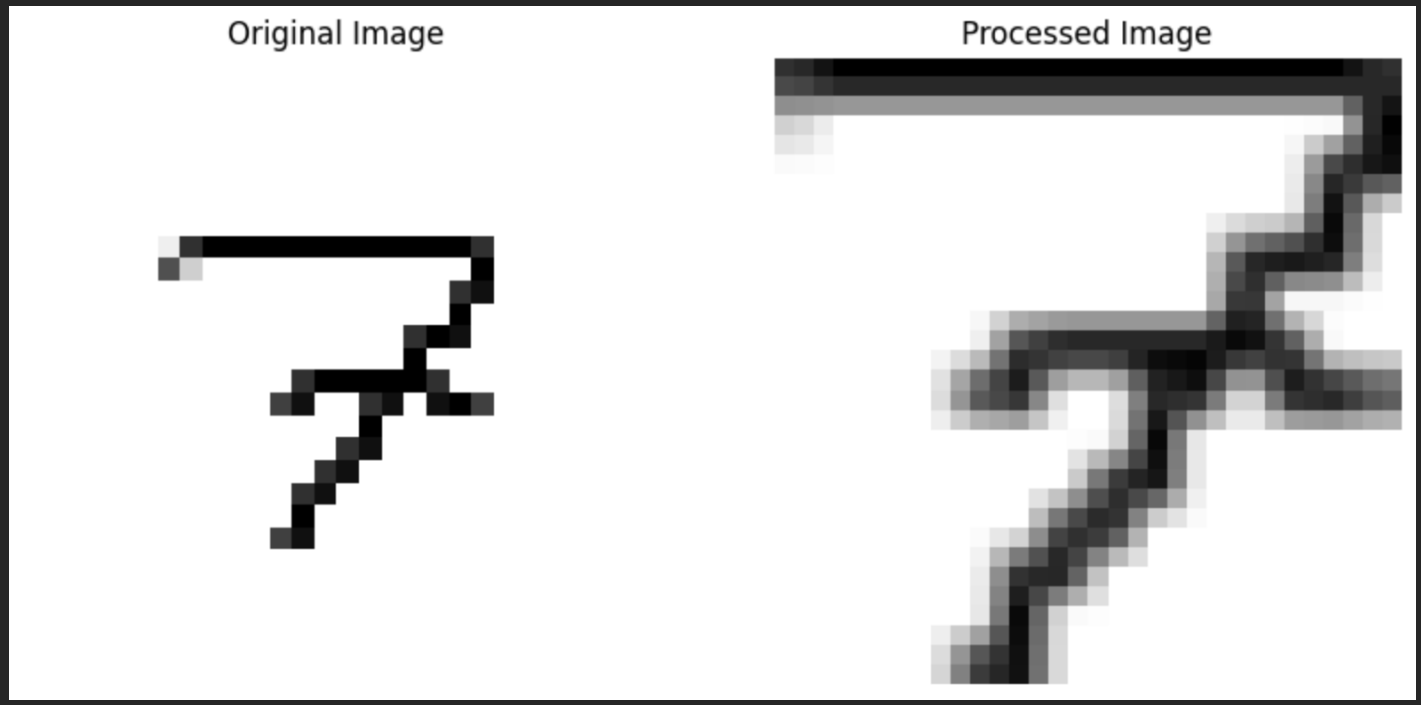
- “NIST Handprinted Forms and Characters - NIST Special Database 19”
- National Institute of Standards and Technology
- https://catalog.data.gov/dataset/nist-handprinted-forms-and-characters-nist-special-database-19-0f025
- tausende Bilder zu jedem Buchstaben
- Keine kompletten Wörter

Datenaufbereitung

- Verkleinern

  Pro: schnell

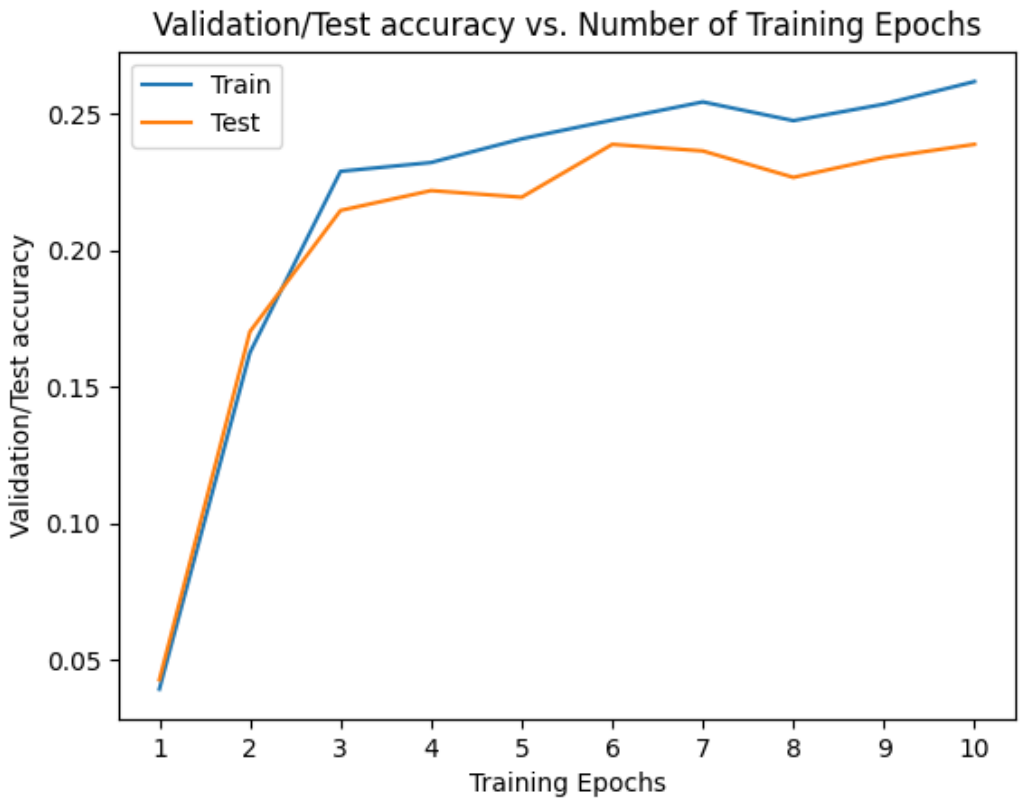
  Zu klein für Architekturen
- Zuschneiden
- Quadratisches Bild erstellen



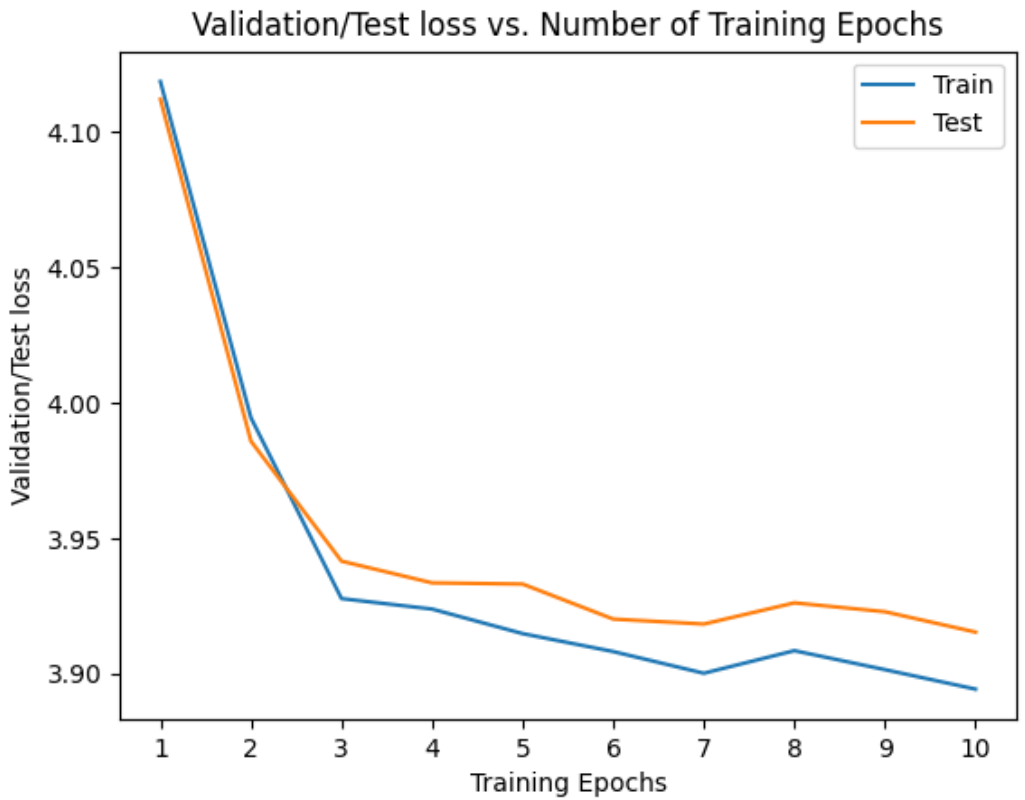
Ergebnisse

LeNet5 (32px)

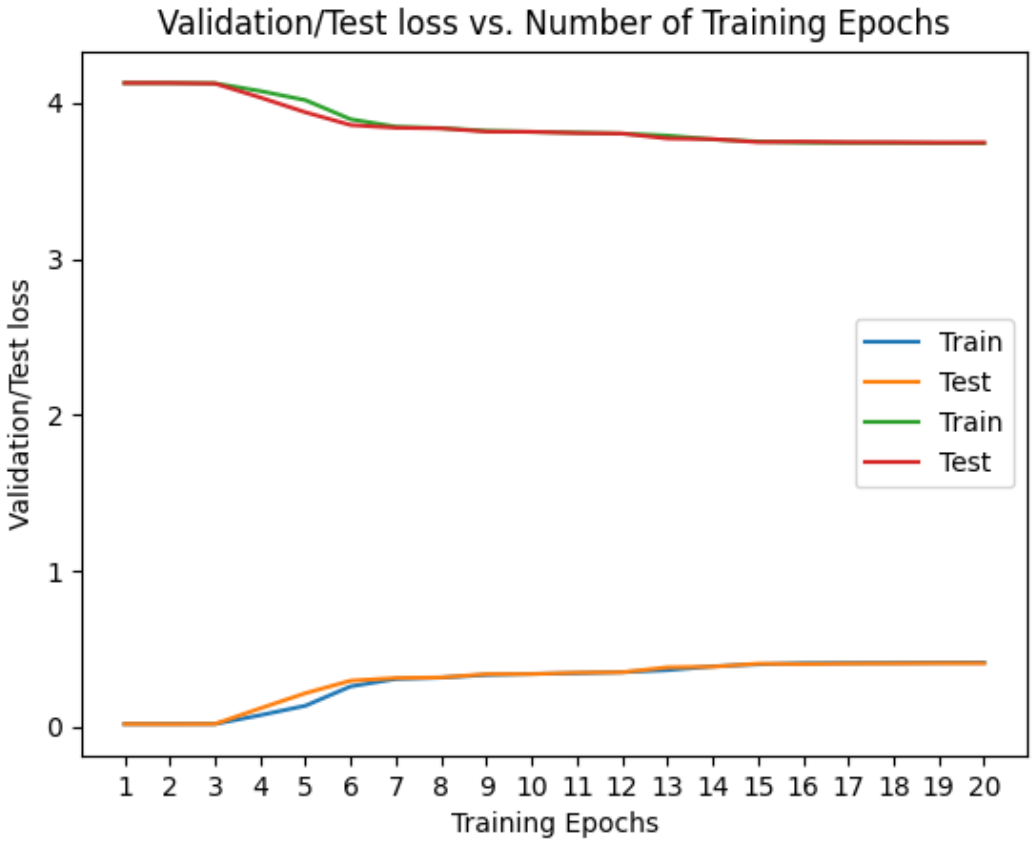
Accuracy



Loss



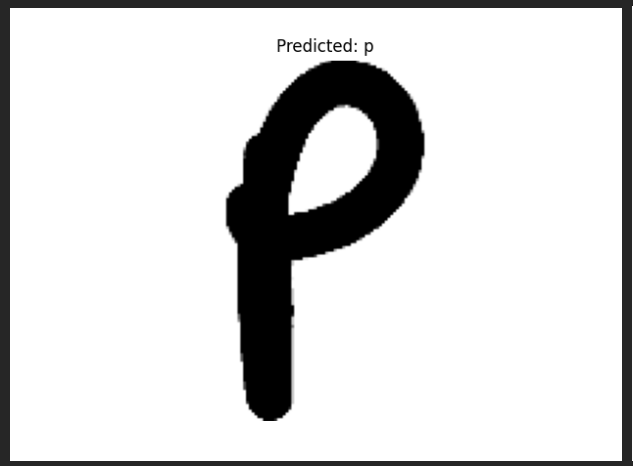
AlexNet (32px)

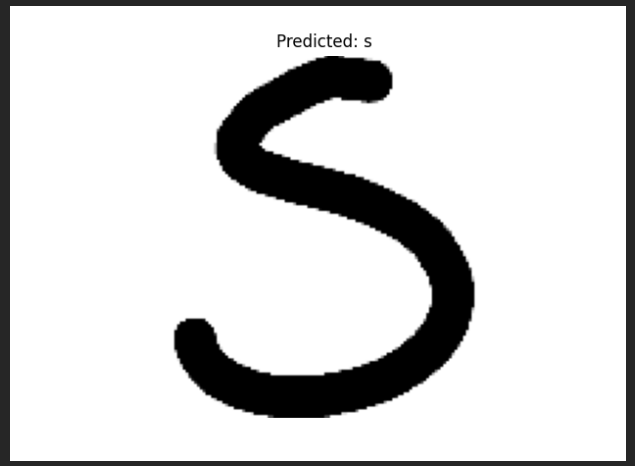


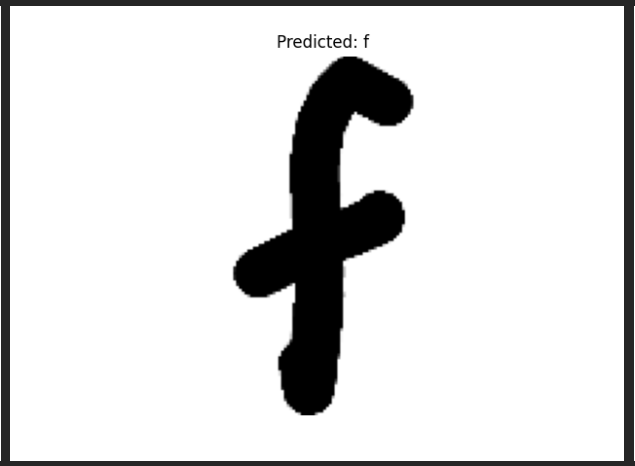
ResNet50 (32px)

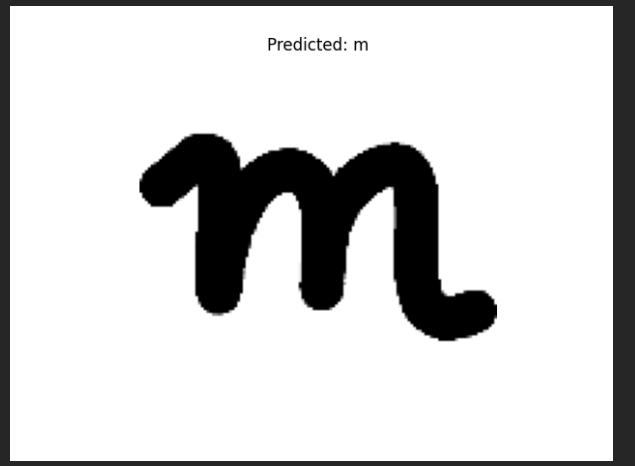
ResNeXt50 (128px)

ResNeXt50: gute Vorhersagen

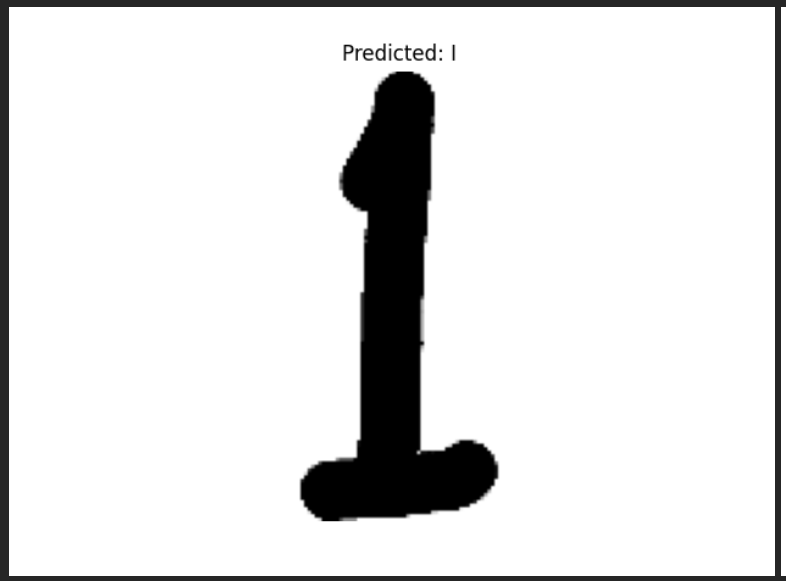


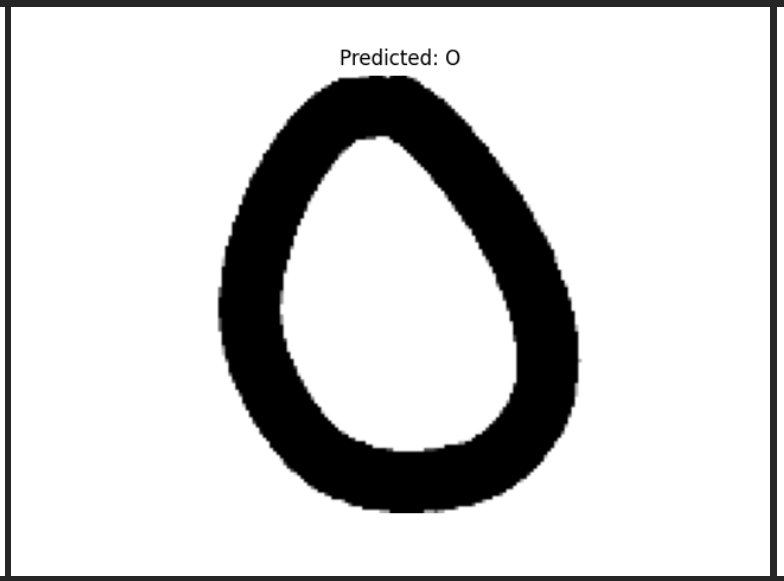


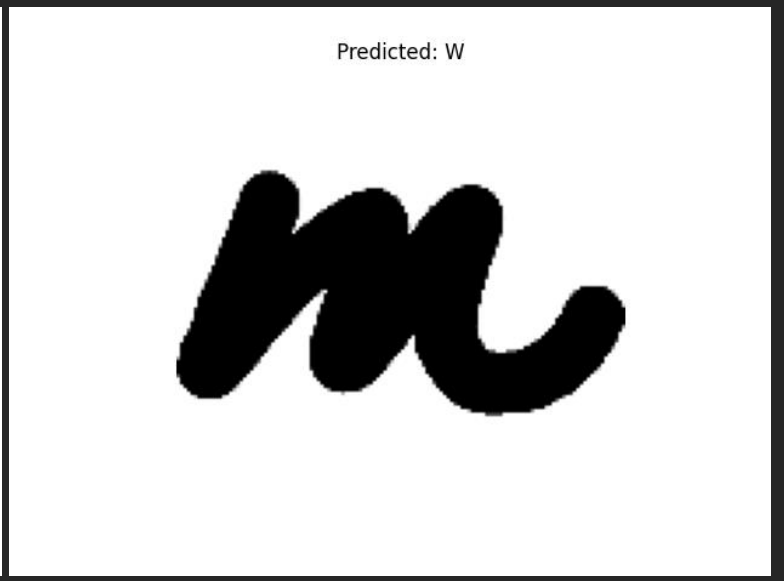




ResNeXt50: Verwechslungen



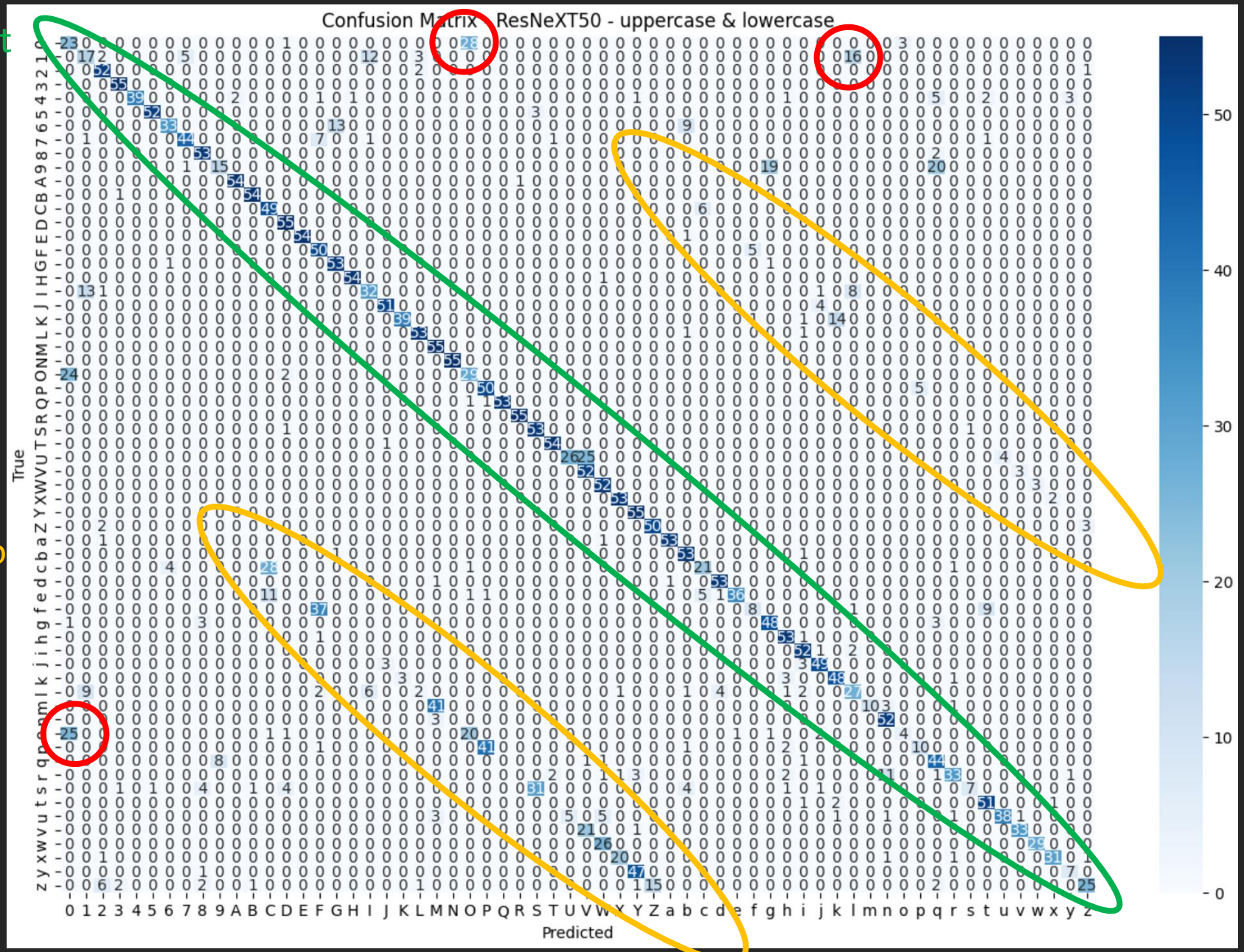




Richtig gewertet

ResNeXt50: Confusion Matrix

Groß-/Kleinbuchstaben verwechselt



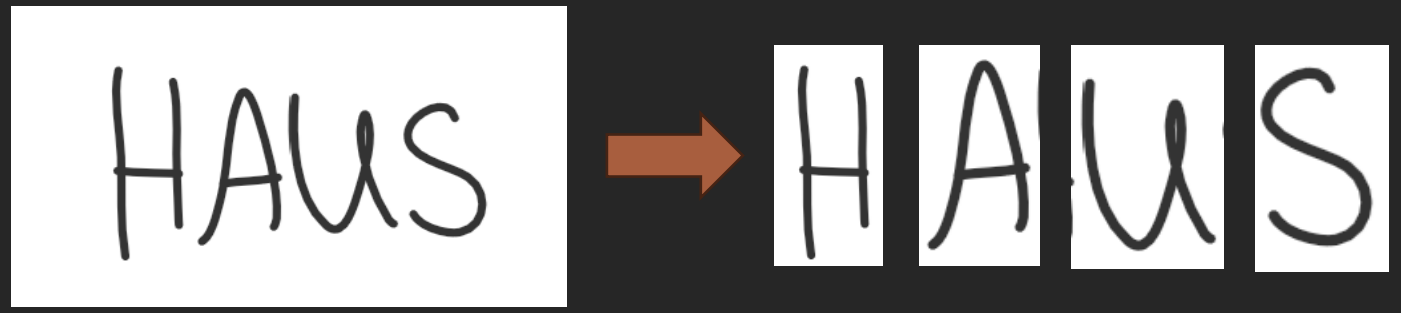
ResNeXt50

	precision	recall	f1-score	support
0	0.32	0.42	0.36	55
1	0.42	0.31	0.36	55
2	0.80	0.95	0.87	55
3	0.93	1.00	0.96	55
4	1.00	0.71	0.83	55
5	0.98	0.95	0.96	55
6	0.87	0.60	0.71	55
7	0.88	0.80	0.84	55
8	0.84	0.96	0.90	55
9	0.65	0.27	0.38	55
A	0.96	0.98	0.97	55
B	0.96	0.98	0.97	55
C	0.55	0.89	0.68	55
D	0.86	1.00	0.92	55
E	1.00	0.98	0.99	55
F	0.51	0.91	0.65	55
G	0.79	0.96	0.87	55
H	0.98	0.98	0.98	55
I	0.63	0.58	0.60	55
J	0.93	0.93	0.93	55
K	0.93	0.71	0.80	55
L	0.87	0.96	0.91	55
M	0.53	1.00	0.70	55
N	1.00	1.00	1.00	55
O	0.36	0.53	0.43	55
P	0.54	0.91	0.68	55
Q	1.00	0.96	0.98	55
R	0.98	1.00	0.99	55
S	0.61	0.96	0.75	55
T	0.95	0.98	0.96	55
U	0.84	0.47	0.60	55
V	0.53	0.95	0.68	55
W	0.60	0.95	0.73	55
X	0.71	0.96	0.82	55
Y	0.51	1.00	0.67	55
Z	0.77	0.91	0.83	55
a	0.98	0.96	0.97	55
b	0.76	0.96	0.85	55
c	0.66	0.38	0.48	55
d	0.91	0.96	0.94	55
e	0.97	0.65	0.78	55
f	0.57	0.15	0.23	55
g	0.70	0.87	0.77	55
h	0.84	0.96	0.90	55
i	0.83	0.95	0.88	55
j	0.84	0.89	0.87	55
k	0.74	0.87	0.80	55
l	0.50	0.49	0.50	55
m	1.00	0.18	0.31	55
n	0.76	0.95	0.85	55
o	0.57	0.07	0.13	55
p	0.67	0.18	0.29	55
q	0.57	0.80	0.67	55
r	0.87	0.60	0.71	55
s	0.88	0.13	0.22	55
t	0.81	0.93	0.86	55
u	0.90	0.69	0.78	55
v	0.89	0.60	0.72	55
w	0.91	0.53	0.67	55
x	0.91	0.56	0.70	55
y	0.64	0.13	0.21	55
z	0.83	0.45	0.59	55
accuracy			0.75	3410
macro avg	0.78	0.75	0.73	3410
weighted avg	0.78	0.75	0.73	3410


~~Buchstaben~~
Wörter

Prinzip

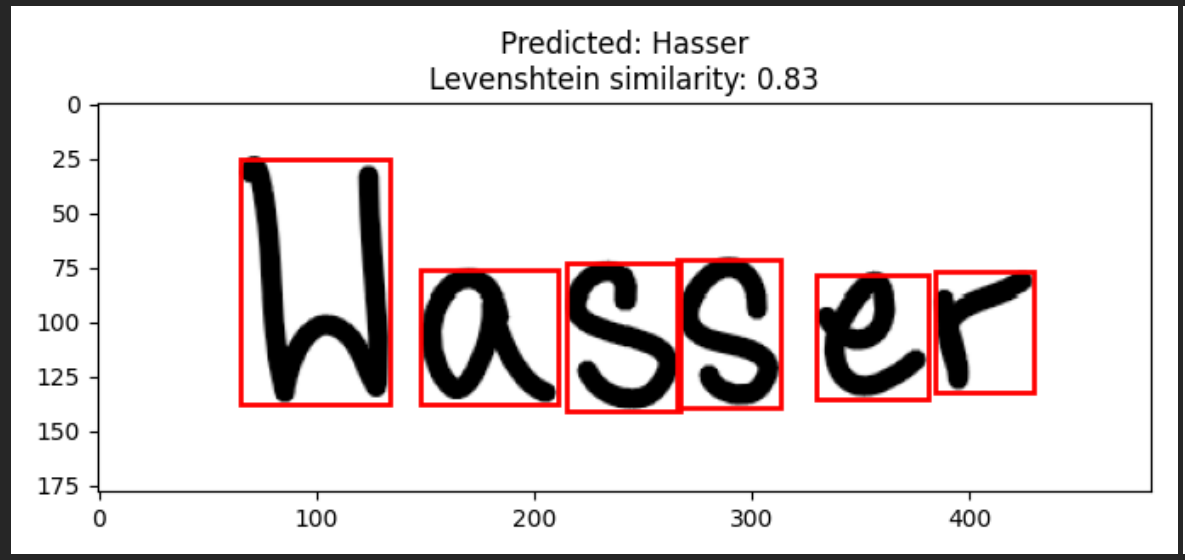
- Ein Wort besteht aus mehreren Buchstaben
- Jeder Buchstabe ist eine zusammenhängende Fläche
- -> Modell auf jede zusammenhängende Fläche anwenden

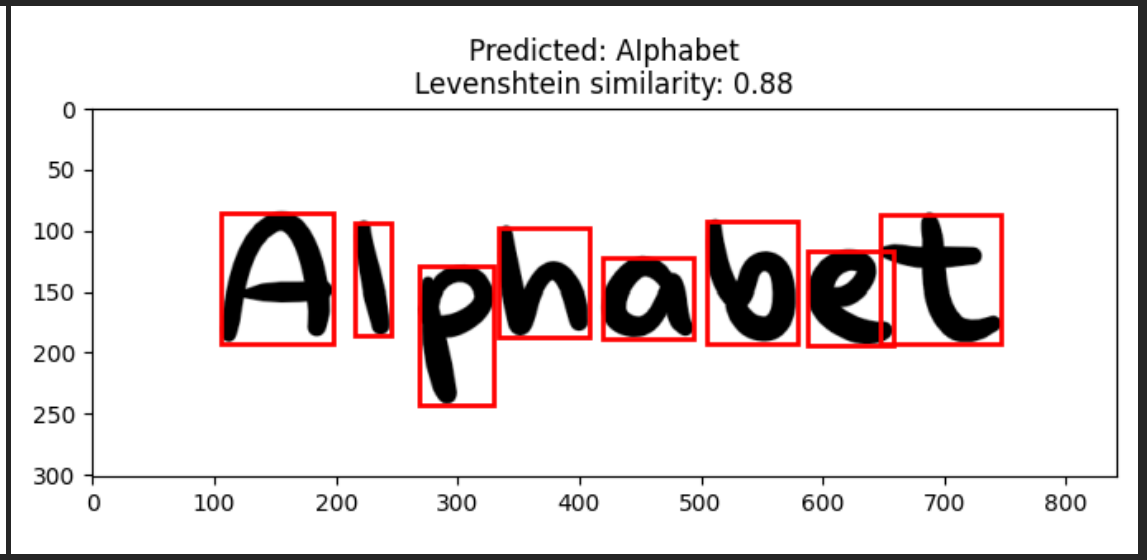


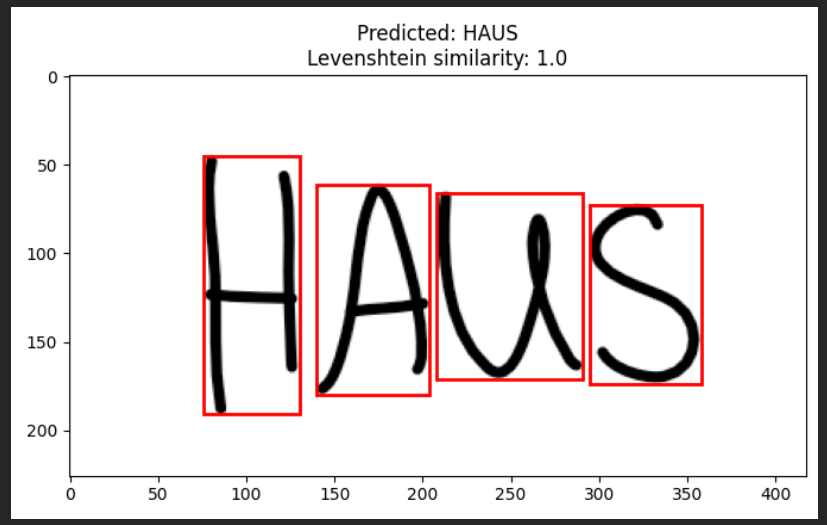
Levenshtein Distanz

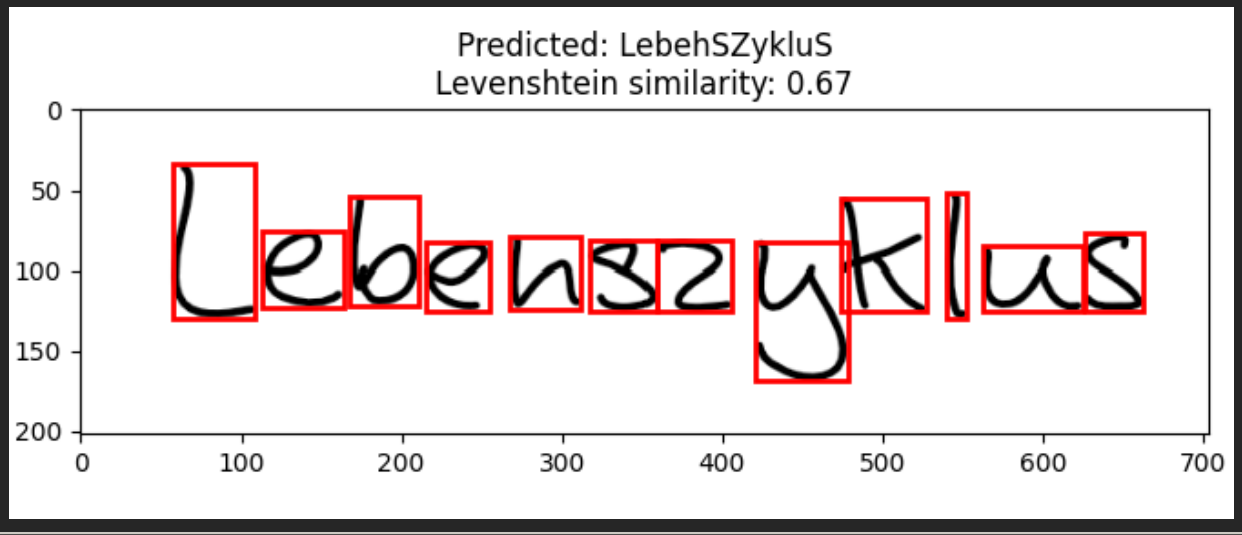
- „minimale Anzahl der notwendigen Änderungen, um zwei Zeichenketten aneinander anzugleichen“
  (https://www.bigdata-insider.de/was-ist-die-levenshtein-distanz-a-502eb7581b6879ee988a93b214821689/)
- Einfügen, Löschen, Ersetzen

ResNeXt50: Wörter

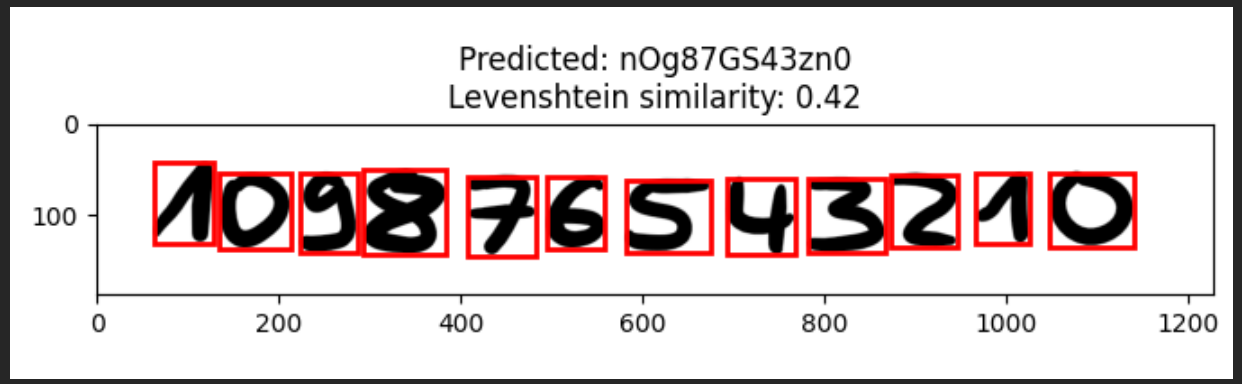


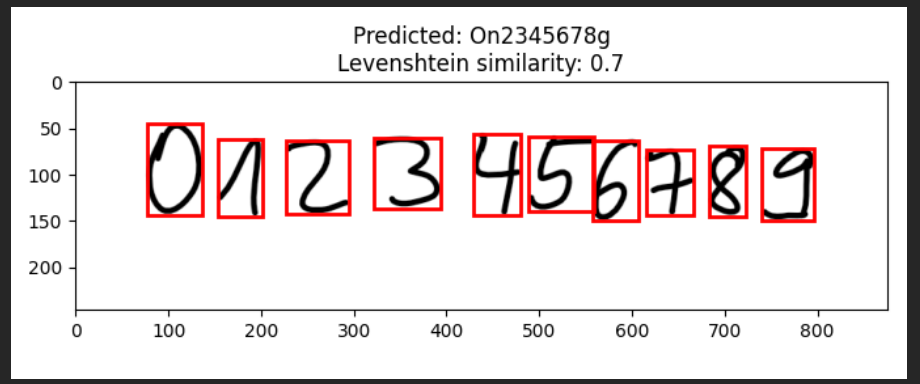


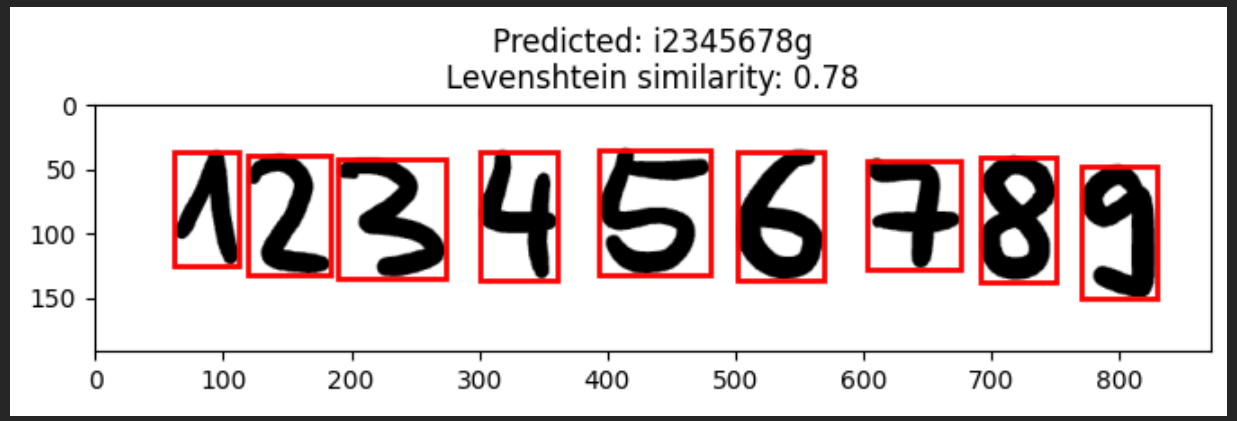




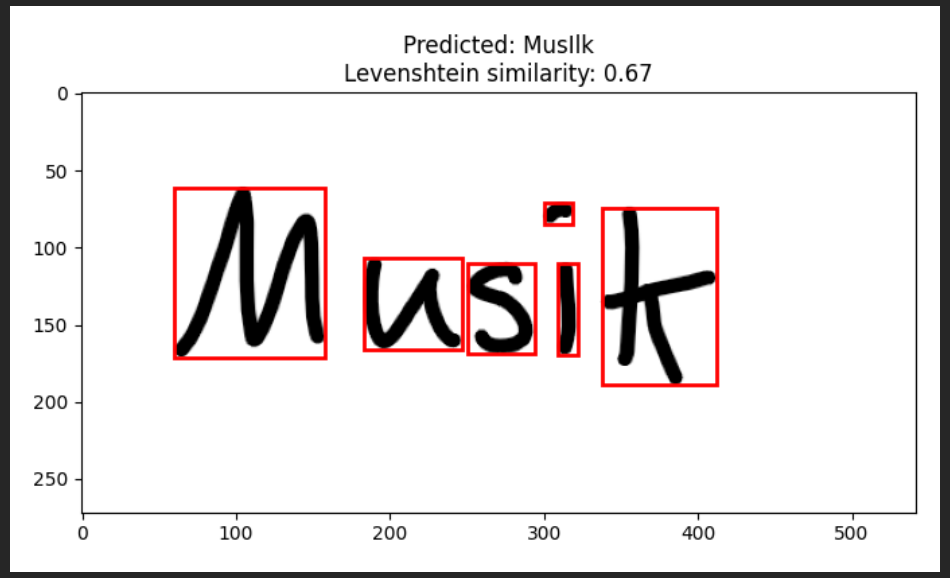
ResNeXt50: Zahlen

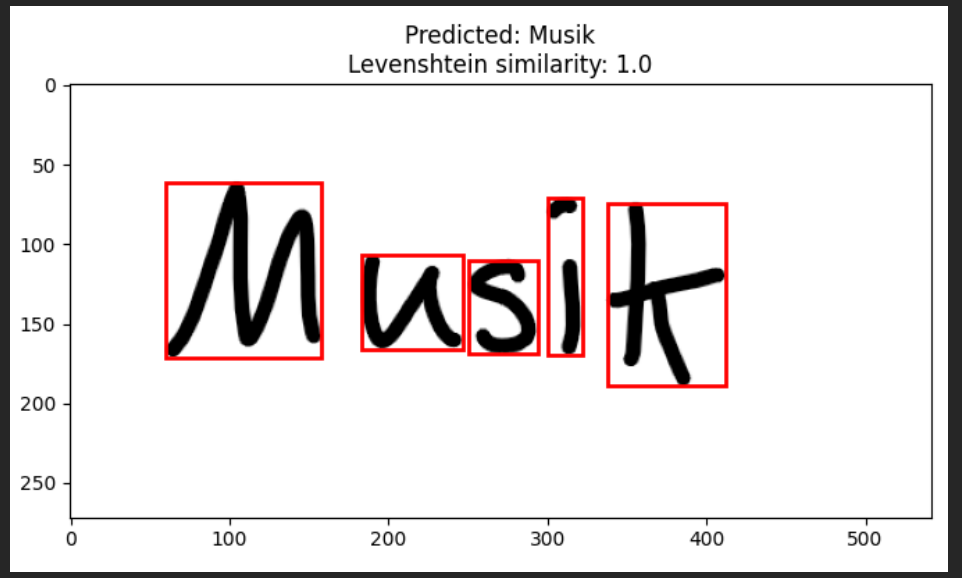






Der Buchstabe „i“

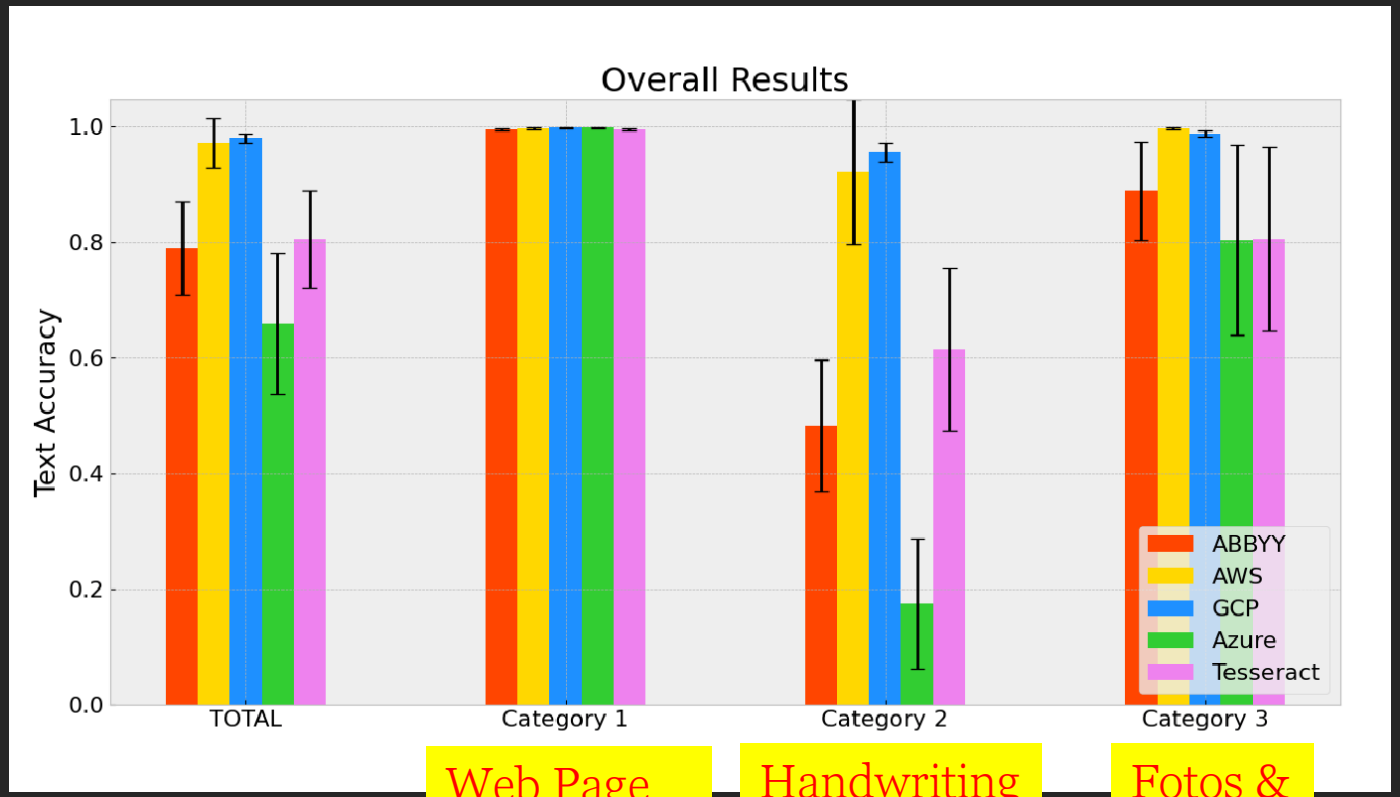




- Levenshtein Distanz gesamt: 0.73

Vergleich

State of the art



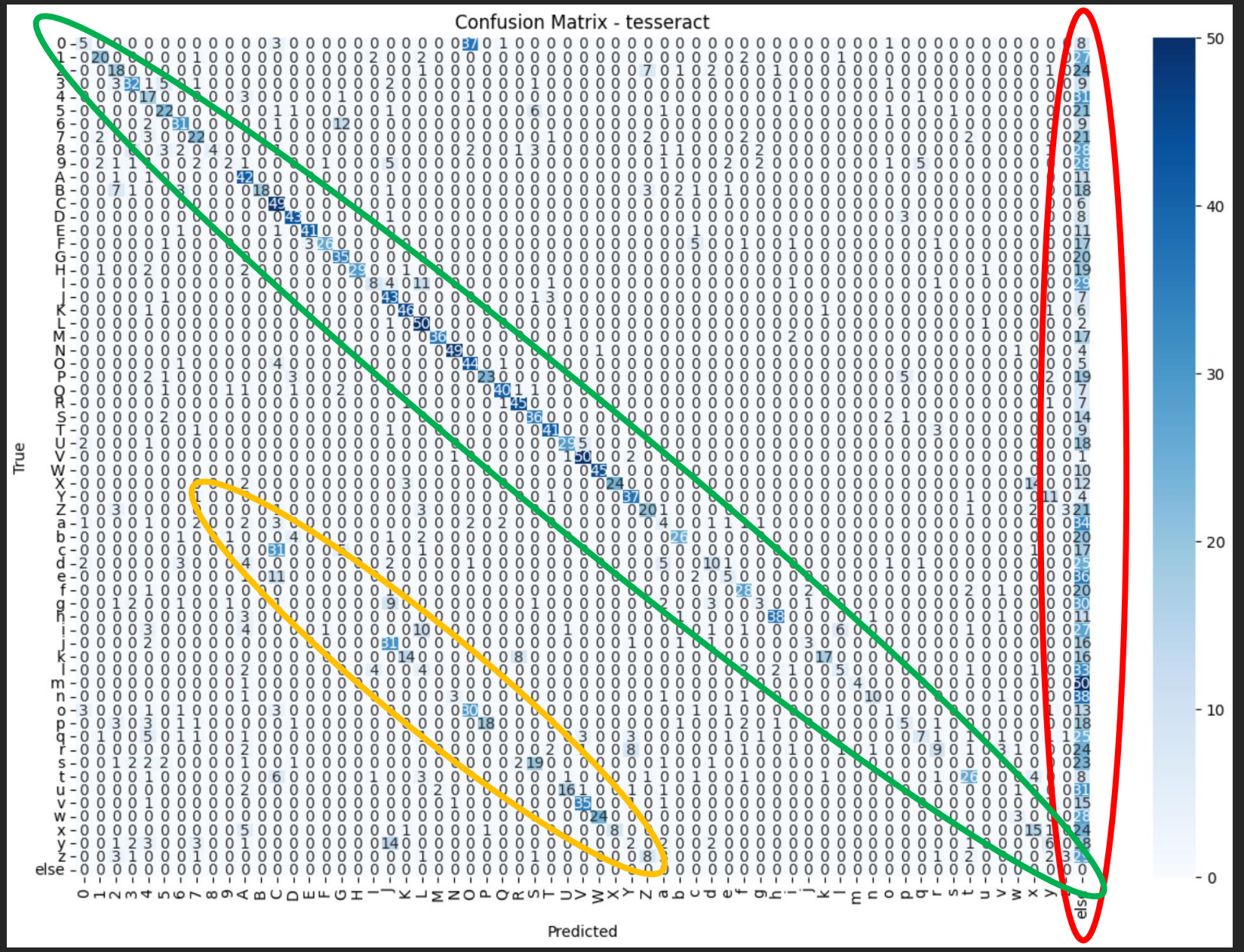
https://research.aimultiple.com/ocr-accuracy/

Web Page Screenshots

Handwriting

Fotos & Scans

Konfusions matrix Tesseract



Vergleich

Variante	Precision	Recall	F1-Score	Levenshtein
Tesseract	0.55	0.39	0.41	0.7
ResNeXt50	0.78	0.75	0.73	0.73

Fazit

- Einzelne Buchstaben vs komplettes Wort

  Weiterer Ansatz: Objekterkennung

  Auch zusammengeschriebene Buchstaben

- Mein Modell erkennt getrennte Buchstaben mit selbst implementierter Bounding Box

  Probleme mit „i“, „ä“, „ö“, „ü“

  Abhängig vom Datensatz

- ResNeXt50 am besten
- Hochauflösende Bilder langsamer, aber besser
- Ziel erreicht