Handwritten → LaTeX

Ole Kröger

IWR, Heidelberg University

Objectives

Recognize handwritten math symbols and combine them to a LaTeX-formula

- Generate handwritten formula by single symbols
- Detection and classification of symbols
- Generating a LaTeX-sequence

Data

- $> 350,000$ images of 82 different math symbols
- Combining 23 different symbols to easy formula
- Normalization of symbols
  - Removing the border
  - Scale to at most 40×40
  - Center the mass in a 48×48 image
  - Subtract mean and divide by standard deviation

Figure 1: Generated formula

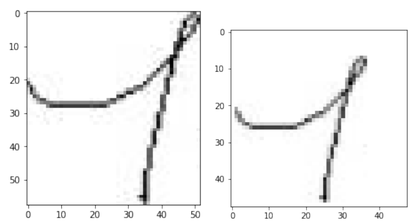

Figure 2: Crop and centered single math symbol

Approaches

Image classification and sliding window

- Image classification using CNN
- Sliding window for object detection
- Very slow
- Low accuracy due to prediction if overlapping (no false examples)

Object recognition in one step

- Detection and classification using Tensorflow API
- Training phase too long
- Too complicated for this problem

Final approach

Image classification using CNN
OpenCV for object detection

- 3 Layer CNN trained on $> 100,000$ images
- Classification with $> 99\%$ accuracy
- OpenCV contour finding and bounding boxes
- Combining overlapping bounding boxes
- fast and reasonable for this simple data

Sequence 2 Sequence Model

- Use predictions and positions
- Simple greedy sequence to sequence model
- Accuracy around 85%

Pipeline

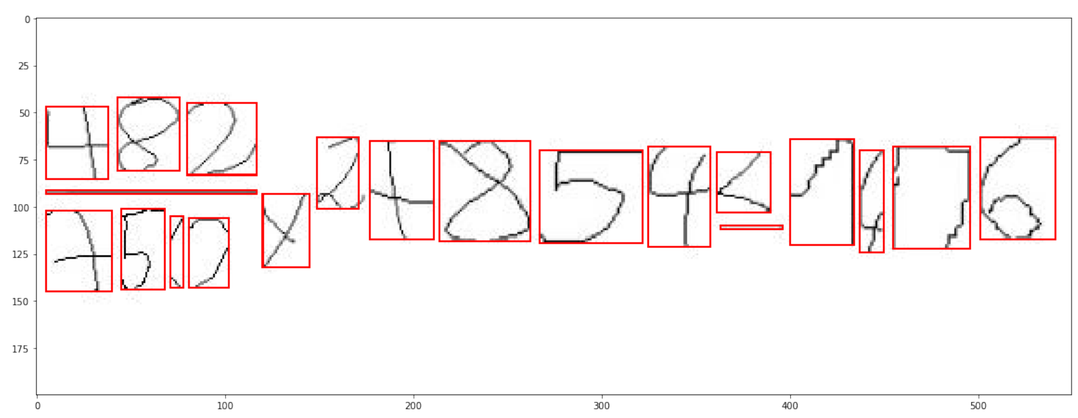

Figure 3: Find single bounding boxes using OpenCV

Use predictions and positions of bounding boxes (relative and absolute) to generate a sequence.
Training data:

- 30.000 images of simple one level formula as shown above
- 30.000 images of fractions in a simple formula

$$\frac{482}{750}y^2 + 854 \leq 1406$$

Results

- $\approx 85\%$ of the symbols in the sequences are correct
- only $\approx 50\%$ of test sequences are completely correct

Conclusion

Combination of three steps

- Detection:
  Relatively good for black symbols on white background but symbols like $\geq$ and $=$ get two bounding boxes.
- Classification:
  Very good classification accuracy using an easy to train 3 Layer CNN.
- Generating a LaTeX-sequence:
  Slower to train and classification accuracy isn't as good.
  Possible improvements:
  - Seqence model with attention
  - Beamsearch
  - More training data

References

[1] Xai Nano.
Kaggle mathsymbols.
Published online at `https://www.kaggle.com/xainano/handwrittenmathsymbols`, 2016.
Accessed: 07/07/2018.

[2] Jeffrey Ling Alexander M. Rush Yuntian Deng, Anssi Kanervisto.
Image-to-markup generation with coarse-to-fine attention.

Published: 13/06/2017.