HM Hochschule München University of Applied Sciences

Bachelor Thesis
in Information Systems and Management

Label Extraction from Image via Deep Learning

Johannes Reichle
Matriculation no. 04797218

Supervisor	Prof. Dr. Rainer Schmidt
Date of Submission	XX.XX.2022

Declaration

I hereby certify that I have written this Bachelor Thesis on my own and that I have not used any sources or aids other than those indicated.

Munich, the XX.XX.2022

. . . .
Johannes Reichle

Abstract

Here abstract for Bachelor Thesis.

Contents

Chapter 1

Introduction

1.1 Motivation

Optical Character Recognition (OCR) is the concept of extracting typed, handwritten or printed text from an image. Techniques for this concept have improved a lot due to the advances in the field of Deep Learning [ZJG$^+$20]. Deep Learning is a technology based on Artificial Neural Networks where data is processed in multiple layers to extract complex features and solve a given problem [SM19]. Deep Learning has only caught on in the recent years as the big computational cost has been met by the improvement in computer hardware [PRN$^+$17]. Finding the right solution in the space of deep learning and applying these new capabilities to the use case of extracting information of equipment labels is the focus of this thesis.

1.2 Problem description

The central part of the bachelor thesis is laying a foundation for finding the right approach for the extraction of textual information from images with equipment labels. This includes defining functional requirements such as detecting rotated text but also non functional requirements such as given computational power of mobile devices the solution. These requirements define properties that a solution must have in order to be classified as viable. Thus the discussion of techniques from end-to-end OCR to dividing the process into text detection and text recognition is centered around the requirements which are given by the problem. Critical factors such as the availability of data and the complexity of implementing and training a Deep Learning model also need to be assessed.

Following aspects such as implementing, training, deploying and maintaining a solution in a production environment shall not be subject of this thesis.

1.3 Methodology

The methodology of this Bachelor Thesis can be described as a literature review. As such, the research question guiding the process is most crucial: Which state of the art Deep Learning approaches for OCR are viable for the use case of extracting textual label data from images. The following section describes how relevant literature is identified, analysed and synthesised.

Add: how was problem defined

Searching: search strategy for specific review: → provide reasoning behind choices

- search tearms
- databases
- inclusion & exclusion criteria: year of publification, type of article, journal, research quality

write all decisions down, how is search documented The research is then extended to existing practical solutions for similar practical problems as well as proposed architectures from academic research. documentation of search process and selection, asses quality of search process and selection

Analysis: how is data prepared for analysis analyse problem: find requirements type of information that needs to be extracted: what works, why, not necesseraly how literature review (state of the art) how? top journals of last few years Theoretical knowledge about the models as well as practical information about results for

Synthesis

after review: discuss different approchoaches pros and cons for problem methodogical limitations: practice is always different, . . .

1.4 Expected results

In the following section the structure of the Bachelor Thesis is listed and each chapter's expected result is detailed along with its benefits for the overall objective of producing an overview of state of the art OCR relevant for the problem described in Section 1.2.

In Chapter 2 the theoretical foundation for the Bachelor Thesis which is needed for comprehension of the following chapters is gathered. This includes general principles of Deep Learning and by extension Machine Learning but also of OCR.

In Chapter 3 the problem from Section 1.2 is addressed in more detail. The result shall be a firm understanding of functional and non-functional requirements both on the technical and the business process side. These requirements are the point of focus for the further examination of OCR techniques.

After laying the foundation, in Chapter 4 current research in regards to the identified requirements is examined. The resulting overview can be viewed as a basis for a decision when it comes implementing a practical solution.

Therefore it enables the discussion in Chapter 5. Here not only the results and the availability of a solution but also the methodology of this work are assessed critically.

The conclusion is a summary of the results compared to the expected results detailed in this chapter as well as an outlook for further research into the topic.

‘A major challenge in developing DL systems is the difficulties in estimating the results before a system has been trained and tested’[ABCFB18]

Chapter 2

Theoretical Foundation

2.1 Machine Learning

1. Loss Function / Error Metrics
2. Supervised — Unsupervised / Categorization
3. Optimization techniques: Stochastic-Batch Gradient Descent, GD Momentum, Adam
4. Bias-Variance tradeoff / Overfitting — Underfitting

2.2 Deep Learning

‘ One of the main differences from traditional ma- chine learning (ML) methods is that DL automatically learns how to represent data using multiple layers of abstraction [5], [6]. In traditional ML, a significant amount of work has to be spent on “feature engineering” to build this representation manually, but this process can now be automated to a higher degree. Having an automated and data-driven method for learning how to represent data improves both the performance of the model and reduces requirements for manual feature engineering work [7], [8].’ [ABCFB18]

1. ANN / MLP
    - Architecture → Input, Hidden, Output
    - Feedforward
    - Optimization → Backpropagation, SGD, ADAM, . . .
2. Regularization

3. important architectures
    - CNN
    - RNN
    - Specific foundation architectures for relevant approaches
4. transfer learning: reuse parameters from pretrained models

Deep Learning in Character Recognition Considering Pattern Invariance Constraints [OOK15] Deep Learning: neural network architecture of more than a single hidden layer as opposed to shallow networks Features of deep networks: distributed representation of knowledge at each hidden layer, distinct features are extracted by units or neurons in each hidden layer several units can be active concurrently Each layer extracts moredefined/advanced features → hierarchical representation of features

Common problems with training deep learning

- saturating units
- vanishing gradients
- over-fitting & underfitting

Classification of deep learning architectures

- Generative Architectures:
  not deterministic of class patterns that input belong to → sample joint statistical distribution of data
  unsupervised learning: greedy layer-wise pre-training
  Use auto encoders (generative) when a lot unlabelled but not a lot labelled data → generatively train network and then fine tune with labelled
- Discriminative Architectures:
  required to be deterministic of correlation of input data to the classes of patterns therein
  supervised learning
- Hybrid
  combination of discriminative and generative
  generally pre-trained and discriminately fine-tuned for deterministic purposes

2.2.1 Convolutional Neural Network

Comparative analysis of deep learning image detection algorithms [SDA+21] These layers apply filters to extract patterns from images. The filter moves over the image to generates the output. Different filters recognize different patterns. Initial layers have filters to recognize simple patterns. They become more complex through the layers over time as follows:

Review of Deep Learning Algorithms and Architectures [SM19] Def Neural Network:

- Machine Learning technique that consists of processing units organized in input, hidden and output layers
- the nodes or units in each layer are connected to nodes in adjacent layers
- each connection has weight value
- inputs are multiplied by weight and summed up at each unit
- the sum is used with an activation function (e.g. ReLU, Sigmoid, Tanh, SoftPlus)

2.3 Opical Character Recognition

Deep Learning based OCR [ZJG+20] What is OCR: process of converting images of typed, handwritten or printed text into machine-encoded one includes two sub frameworks: text detection and text recognition (based on position coordinates) **End-To-End also possible** Process can include image processing!!!

no source grid: divides image into parts → each part has own bounding boxes bounding boxes: regressor for box, each bounding box is assigned an anchor box (respective to grid cell) anchor boxes: default ‘shape’ for bounding box

bounding boxes different stages of convolution / 2-d size → different object size to detect

2.3.1 Text detection

subfield of object detection (e.g. YOLOv4 can be used for text)

Detect position coordinates containing text in input image Text detection more challenging

Two object detection methods — CNN-based

- Region-based
  views detection problem as classification problem
  CNN to extract deep features of proposals by selective search → Use SVM to classify with features
  e.g. R-CNN

- single 'look' extract feature maps on entire image
  directly regress bounding boxes on feature maps
  e.g. YOLO — You Only Look Once, SSD — Single Shot Detection

Non CNN-based: DETR

Comparison Object Detection basic algos

Comparative analysis of deep learning image detection algorithms [SDA+21]
YOLO-V3 outperforms SSD and Faster R-CNN
VGG-16 widely used feature generating architecture

Faster-RCNN

A deeper look at how Faster-RCNN works [Gos18] composed of 3 neural nets:

- Feature Network: pre-trained image classification netork → generate good features
- Region Proposal Network:
  - NN with 3 conv layers
  - one layer splits up network to: classification and bounding box regression
  - bounding box regression → bounding boxes are region of interes (ROI) that might contain an object
- Detection Network: take input from previous nets, generate final class and bounding box, 4 fully connected, 2 stacked common layers shared by classification and bounding box regression layer

Deep Learning in Character Recognition Considering Pattern Invariance Constraints [OOK15] Neural networks can learn features of task on which they are designed and trained Neural networks better than other approaches (e.g. template matching, syntactic analysis) → NNs can learn and adapt to moderate variations (e.g. translation, rotation, scaling, noisy patterns)

2.3.2 Text Recognition

Recognize text based on position coordinates
character based or word based

2.3.3 End-To-End

Chapter 3

Problem analysis

This chapter entails an analysis of the problem which is the research question's foundation. As the quality of requirements ultimately determines the quality of the literature review this chapter is crucial. The basic problem can be described as extracting textual data from images with the following features; The relevant text information is 'framed' by the label and is structured in key-value pairs like 'Serial-No. 1234567', the text can have different spacing inside the label.

Requirements

- problem desc / functional
  - extract printed textual data from image
  - extract with same spacing || simply extract and keep semantical spacing
  - pattern match semantic value of text — serial number, ...
  - where to put extracted text? html, pdf?
  - accuracy/reliability: which metric to use (considered functional for ML [VB19])
- non-functional: restricts degree of freedom for solution for functional requirements
  - efficiency → because of mobile phone
  - portability → kind of mobile phone, also usable on e.g. MacBook
  - discuss: conditions for data preparation, definitions of outlieres, derived data
  - different mobile phone cameras → resolutions, ...

  - mobile phone computational capabilities → performance, GPU/ANE
  - no internet access
  - image properties
    - only printed text
    - different fonts, font sizes
    - different spacings
    - image quality threshhold: fuzzy, blurry, pixels
    - rotation
  - complexity of training
  - needed data set

- Data Requirements → training, tuning, testing: 'Based on our interviews, we would add "training data needs specified and validated requirements like code"' [VB19]
  - Quantity: constraints on amount of data necessary
  - Quality: better quality → better application → clean and augment data is important
most important quality dimensions: completeness, consistency, correctness

Inwiefern auch requirements fürs training? data and code dependencies complexity of training needed data set pipeline definition or just recommendation for model?

tradeoff between accuracy and computational cost delimination here: The focus of this Bachelor Thesis is limited to ...

- implementation and respective aspects like cost, complexity of large amounts of data
- training
- deployment
- maintaining

Def Requirements [ZJJB+14]

- functional: specify functions that a system or system component must deliver to users [noa98]

- non-functional: other requirements that play important role in shaping target system, defining the development process and managing the development project [KS98, CdPL09]

challenges from DL perspective [ABCFB18]

- Development
  - 'It is challenging to provide a sample that includes all the edge cases that may exist in the full dataset. Also, as the external world is dynamic and changes over time, new edge cases will continue to appear later in time.'
  - 'A major challenge in developing DL systems is the difficulties in estimating the results before a system has been trained and tested. '
  - reproducible results dependent on components like hardware, platform, . . .
- Production
- Organizational

Requirements Engineering for Machine Learning [VB19] 'In addition, a recent survey suggests that Requirements Engineering (RE) is the most difficult activity for the development of ML-based systems [2].'

3.1 Functional Requirements

3.2 Non-Functional Requirements

3.3 Business Process

When determining whether automisation is an improvement four aspects have to be examined. These are time, costs, quality and flexibility. The aspects build a quadrangle that is based on the optimizing trade-off between the factors [DLRMR13].

Without software supporting the task of reading the name of the picture and typing it into the system, can take long seconds, whereas a trained Deep Learning model could complete the task in a mere instant. Therefor automisation via Deep Learning should improve the efficiency of the process when compared to manually reading and typing the information off the image.

Training costs for a Deep Learning model are very high due to the computing intensive backpropagation algorithm that tunes the network to the data. But the usage cost is low. For manual labor the opposite is the case as training a person to type in a label is done quickly and labor costs are high in comparison to the expenses for running the model.

Both Deep Learning models and human labor are not 100% accurate. The question is whether the model can be as accurate or even better than its human counterpart. This is especially interesting when it is applied in the real world where it might have to do good in subpar situations. An example is bad image quality.

Flexibility is concerned with how well a process can adjust to changing requirements. A set of new equipment names that have to be included can pose a problem to a Deep Learning model because it is not trained for the new data. A human on the other hand should not have any problems in this regard.

The main concern for the solution's efficacy is whether it is accurate enough. Therefor this work focuses on this aspect in particular.

Chapter 4

Current Research

no transformers → self-attention mechanism is too computationally expensive???

model-pruning → remove connections for better performance

4.1 Selection

4.2 Analysis

'The great advances that have been made in fields such as computer vision and speech recognition, have been accom- plished by replacing a modular processing pipeline with large neural networks that are trained end-to-end [37]. In essence, transparency is traded for accuracy. This is an unavoidable reality.'[ABCFB18]

include Pipeline differences

Two models that can be used in conjunction **detection** [Beo21b]
uses RetinaNet structure [LGG+18]
applies techniques from textboxes++ [LSB18]

character recognition [Beo21a]
needs cropped text area as input
uses CRNN [SBY15] → end-to-end learning, LSTM fir arbitrary length of input and output, no need to apply detection and cropping to each single character

Open Source OCR engine [Smi07]

- uses Deep Learning (found c++ code for layers in repo)
- Processing in step-by-step pipeline, some unusual stages
  1. Line and Word finding
  1.1. Line finding

1.2. Baseline Fitting
1.3. Fixed Pitch Detection and Chopping
1.4. Proportional Word Finding
2. Word Recognition
2.1 Chopping Joined Characters
2.2 Accociating Broken Characters
3. Static Character Classifier
3.1 Features
3.2 Classification
3.3 Training Data
4. Linguistic Analysis
5. Adaptive Classifier

Performs poorly with unstructured text with significant noise
An Efficient and Accurate Scene Text Detector [ZYW+17]
SOFT: Softmax-free Transformer with Linear Complexity [LYZ+21]
Generative Pretraining from Pixels [?]

- unsupervised representation learning (approach transfered from NLP)
- training of sequence Transformer to auto-regressively predict pixels without incorporating knowledge of 2D input structure
- Active part: GPT-2 scale model learns image representations and performs extremely well even when compared to supervised models

Learning High-Precision Bounding Box for Rotated Object Detection via Kullback-Leibler Divergence [YYY+21]

- Deductive approach to rotated object detection
- box is 'translated' to 2D-Gaussian → KLD with prediction and true gaussian as Loss
- LIMIT: cannot be directly applied to quadrilateral detection

DP-SSL: Towards Robust Semi-supervised Learning with A Few Labeled Samples [XDZZ21]

- Semi-supervised learning:
  - provides way to leverage unlabeled data by pseudo labels
  - performs poorly and unstable when size of labeled data is very small (low quality of pseudo labels)

- Data programming:
  - paradigm for the programmatic creation of training sets
  - existing methods rely on human experts to provide initial labeling functions (LF)
- DP-SSL
  - multiple-choice learning (MCL) based approach to automatically generate labeling functions
  - scheme to generate probabilistic labels for unlabeled data

4.3 Synthesis

which aspects to compare? quantitative, qualitative

Chapter 5

Discussion

5.1 Results

5.2 Method reflection

Challenges DL[ABCFB18] Note that actual experiments with models have to be done Problem: different papers have different components → Hardware, Platform, Source Code, Configuration → studies can't really be compared

5.3 Outlook

Chapter 6

Conclusion

Appendix A

References

List of Figures

List of Tables

Bibliography

[ABCFB18] Anders Arpteg, Björn Brinne, Luka Crnkovic-Friis, and Jan Bosch. Software Engineering Challenges of Deep Learning. In *2018 44th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, pages 50–59, August 2018.

[Beo21a] Beom. CRNN (CNN+RNN), September 2021. original-date: 2018-01-14T07:52:25Z.

[Beo21b] Beom. Text Detector for OCR, August 2021. original-date: 2019-03-12T05:11:06Z.

[CdPL09] Lawrence Chung and Julio Cesar Sampaio do Prado Leite. On Non-Functional Requirements in Software Engineering. In Alexander T. Borgida, Vinay K. Chaudhri, Paolo Giorgini, and Eric S. Yu, editors, *Conceptual Modeling: Foundations and Applications: Essays in Honor of John Mylopoulos*, Lecture Notes in Computer Science, pages 363–379. Springer, Berlin, Heidelberg, 2009.

[DLRMR13] Marlon Dumas, Marcello La Rosa, Jan Mendling, and Hajo A. Reijers. *Fundamentals of Business Process Management.* Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[Gos18] Subrata Goswami. A deeper look at how Faster-RCNN works, July 2018.

[KS98] Gerald Kotonya and Ian Sommerville. *Requirements Engineering: Processes and Techniques.* Wiley Publishing, 1st edition, 1998.

[LGG$^+$18] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal Loss for Dense Object Detection. *arXiv:1708.02002 [cs]*, February 2018. arXiv: 1708.02002.

[LSB18] Minghui Liao, Baoguang Shi, and Xiang Bai. TextBoxes++: A Single-Shot Oriented Scene Text Detector. *IEEE Transactions on Image Processing*, 27(8):3676–3690, August 2018. arXiv: 1801.02765.

[LYZ+21] Jiachen Lu, Jinghan Yao, Junge Zhang, Xiatian Zhu, Hang Xu, Weiguo Gao, Chunjing Xu, Tao Xiang, and Li Zhang. SOFT: Softmax-free Transformer with Linear Complexity. *arXiv:2110.11945 [cs]*, October 2021. arXiv: 2110.11945.

[noa98] IEEE Standard for a Software Quality Metrics Methodology. *IEEE Std 1061-1998*, pages i–, December 1998. Conference Name: IEEE Std 1061-1998.

[OOK15] Oyebade K. Oyedotun, Ebenezer O. Olaniyi, and Adnan Khashman. Deep Learning in Character Recognition Considering Pattern Invariance Constraints. *International Journal of Intelligent Systems and Applications*, 7(7):1–10, June 2015.

[PRN+17] Moacir Antonelli Ponti, Leonardo Sampaio Ferraz Ribeiro, Tiago Santana Nazare, Tu Bui, and John Collomosse. Everything You Wanted to Know about Deep Learning for Computer Vision but Were Afraid to Ask. In *2017 30th SIBGRAPI Conference on Graphics, Patterns and Images Tutorials (SIBGRAPI-T)*, pages 17–41, October 2017. ISSN: 2474-0705.

[SBY15] Baoguang Shi, Xiang Bai, and Cong Yao. An End-to-End Trainable Neural Network for Image-based Sequence Recognition and Its Application to Scene Text Recognition. *arXiv:1507.05717 [cs]*, July 2015. arXiv: 1507.05717.

[SDA+21] Shrey Srivastava, Amit Vishvas Divekar, Chandu Anilkumar, Ishika Naik, Ved Kulkarni, and V. Pattabiraman. Comparative analysis of deep learning image detection algorithms. *Journal of Big Data*, 8(1):66, December 2021.

[SM19] Ajay Shrestha and Ausif Mahmood. Review of Deep Learning Algorithms and Architectures. *IEEE Access*, 7:53040–53065, 2019. Conference Name: IEEE Access.

[Smi07] R. Smith. An Overview of the Tesseract OCR Engine. In *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, volume 2, pages 629–633, September 2007. ISSN: 2379-2140.

[VB19] Andreas Vogelsang and Markus Borg. Requirements Engineering for Machine Learning: Perspectives from Data Scientists. In *2019 IEEE 27th International Requirements Engineering Conference Workshops (REW)*, pages 245–251, September 2019.

[XDZZ21] Yi Xu, Jiandong Ding, Lu Zhang, and Shuigeng Zhou. DP-SSL: Towards Robust Semi-supervised Learning with A Few Labeled Samples. *arXiv:2110.13740 [cs]*, October 2021. arXiv: 2110.13740.

[YYY$^+$21] Xue Yang, Xiaojiang Yang, Jirui Yang, Qi Ming, Wentao Wang, Qi Tian, and Junchi Yan. Learning High-Precision Bounding Box for Rotated Object Detection via Kullback-Leibler Divergence. *arXiv:2106.01883 [cs]*, October 2021. arXiv: 2106.01883.

[ZJG$^+$20] Zhenyao Zhao, Min Jiang, Shihui Guo, Zhenzhong Wang, Fei Chao, and Kay Chen Tan. Improving Deep Learning based Optical Character Recognition via Neural Architecture Search. In *2020 IEEE Congress on Evolutionary Computation (CEC)*, pages 1–7, July 2020.

[ZJJB$^+$14] Didar Zowghi, Zhi Jin, Simone Diniz Junqueira Barbosa, Phoebe Chen, Alfredo Cuzzocrea, Xiaoyong Du, Joaquim Filipe, Orhun Kara, Igor Kotenko, Krishna M. Sivalingam, Dominik Ślęzak, Takashi Washio, and Xiaokang Yang, editors. *Requirements Engineering*, volume 432 of *Communications in Computer and Information Science*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014.

[ZYW$^+$17] Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, and Jiajun Liang. EAST: An Efficient and Accurate Scene Text Detector. *arXiv:1704.03155 [cs]*, July 2017. arXiv: 1704.03155.

Appendix B

Code