Universität Osnabrück
Fachbereich Humanwissenschaften
Institute of Cognitive Science

Bachelorthesis - Expose

Image Segmentation

Sven Groen
970219
Bachelor's Program Cognitive Science

First supervisor: Dr. Ulf Krumnack
Institute of Cognitive Science
Osnabrück

Second supervisor: Manuel Kolmet
IMANOX GmbH
Berlin

Contents

1 Goal and background information

1.1 Cooperation with IMANOX

This project is realized in cooperation with IMANOX. IMANOX is a Berlin-based Startup that developed a smart photo booth for expositions, events and promotions. This photo booth enables customers virtual product placements using augmented and mixed reality. Main features are hand-tracking, digital masks and changing virtual backgrounds.

1.2 Virtual backgrounds

Currently, the photo booth is using a built in Kinect RGB-D camera that measures the distance of objects by casting infrared illumination onto the scene and indirectly measuring the time it takes to travel back to the camera. The camera struggles with correctly predicting the depth in certain situations. Pixels are rendered as invalid and no depth information is provided. The reasons for this are numerous. Pixels might get under saturated (signal is not strong enough) or over saturated (signal is too strong). Other artifacts occur due to the geometry of the scene. The sensors of the camera might receive signals from multiple locations in the scene, leading to an ambiguous depth. Especially around the edges and borders of objects pixels contain mixed signals from fore-and background leading to blurred outlines. In the current version of the photo booth alpha values (0 to 1) are calculated based on the data from the depth sensor. This is done by applying an alpha matting algorithm [1] to the raw depth data. Objects in the foreground receive high alpha values and the background is considered to have an alpha value of 0, making it transparent. In this way the background can be virtually replaced without affecting the objects in the foreground. Due to the described inaccurate data that is given by the depth sensor the result is of low quality. The edges and borders of the objects or people in the scene are not sharp and often misclassified. Especially for small / thin objects, e. g. hair, the camera hardly recognizes it and parts of the hair are therefore considered as background and

are also replaced by the virtual background. For more detailed information on this issue see `https://docs.microsoft.com/de-de/azure/Kinect-dk/depth-camera` [2].

1.3 Goal of the thesis

The goal of this bachelor thesis is to improve the quality of the semantic segmentation of the current IMANOX photo booth using machine learning techniques. The machine learning model will be one of the artificial neural network architectures (see Section 2.1 for details). This model will be altered and changed towards the needs of the project and will be trained with custom training data (see Section 3). It will be investigated whether high quality training data, which will be generated during the project, improves the visual result of the segmentation. When looking at datasets that are very commonly used for semantic segmentation, such as COCO-dataset [3], one can notice that the outlines of the target labels are very rough and not detailed. This might function as a bottleneck for a more detailed segmentation, one that would be able to grasp details such as hair. The exact plan of how this detailed segmentation is achieved is not set yet. Since the problem boils down to identifying the depth of objects in the scene, this thesis might also deal with the question of how additional depth information could be used to improve the result of models than usually just require two-dimensional images. How this depth information is best gathered will be a subject of the thesis itself. Lastly, it has been shown [4, 5] that using Convolutional Long-Short-Term-Memory (Conv-LSTM [6]) layers help to improve the performance of already existing architectures. The thesis will also explore whether adding these Conv-LSTM layers helps predicting the segmentation of future frames, given previous frames and whether the increase accuracy justifies the additional inference time.

2 Segmentation

Szeliski [7] refers to image segmentation as "the task of finding groups of pixels that 'go together' " (p. 237). In the following, semantic segmentation refers to a pixel-wise classification of an image [8]. In the classical image classification tasks the task is to name the objects that can be seen in an image. Semantic segmentation extends this problem further. Each pixel in an image is assigned to one category label given a set of categories. However, individual instances of an object in one image are not differentiated. When individual instances in an image should be recognized, object detection is necessary. For single objects this would be a classification + localization task. Object detection is usually realized by framing the object with a box and assigning a category label to each box. Lastly, there is instance segmentation. Instance segmentation extends the problem of object detection by a pixel-wise classification (similar to semantic segmentation) but with instances being differentiated [8]. See (Figure 2.1) for an overview of the described tasks. Given the goal of this project, only binary semantic segmentation is necessary. Detailed information about which objects are in the scene is not required.

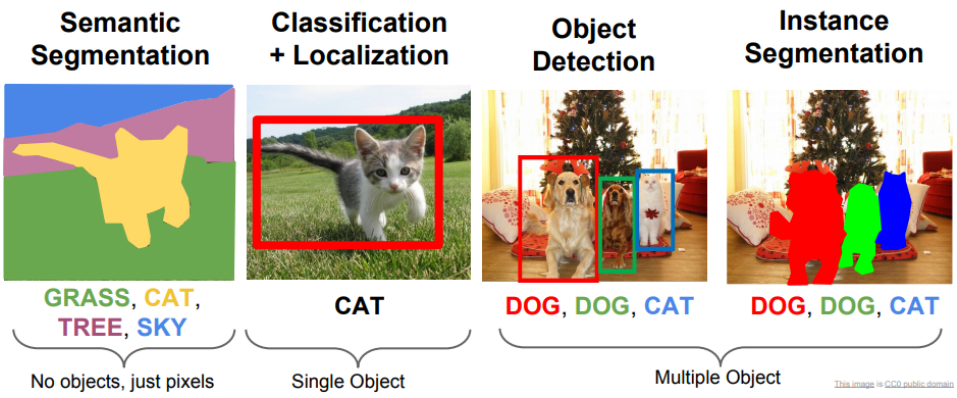

Figure 2.1: Overview of Computer Vision Tasks. (Li *et al.* [9], Slide 17).

2.1 Related work

Segmenting an image into its individual parts is a classical problem of computer vision [7]. Early approaches involve classical methods like threshold detection [10], while modern approaches like k-means clustering [11] improved the results. Deep learning architectures, especially convolutional neural networks (CNNs) [12], have lead to an improvement in performance whereas classical methods have seem to have reach a plateau (Figure 2.2). Long *et al.* [13] have been the first to propose a CNN architecture where a pixel-wise supervised training was achieved. This was done by upsampling the class prediction layer to the input image size, leading to a pixel-wise classification, a Fully Convolutional Network (FCN). Following papers proposed different architectures. A "Deconvolutional Network" with special unpooling and deconvolution operations[14]. The SegNet model uses a similar Encoder-Decoder architecture by using pooling indices to upsample the image [15]. ICNet [16] was able to perform semantic segmentation not only in real time, but also for high quality images (1024x2048 at 30 fps). This was achieved by using a cascade image input of different resolutions. The authors made use of the semantic information from the scaled down images and the details from the high resolution images. Therefore, achieving a "trade-off between efficiency and accuracy" ([16], p.2). Google's approach towards instance segmentation, called Deeplab, has evolved in recent years. The first DeepLab version uses a combination of Deep CNNs with fully connected conditional random fields (CRFs) that tries to grasp the semantic context of the image [17]. The most recent approach, DeepLab V3+, has an Encoder-Decoder structure and was able to show "new state-of-the-art performance on PASCAL VOC 2012 and Cityscapes datasets." ([18], p.14) Moreover, the Google research team has shown that they are able to create an Encoder-Decoder architecture that is very light and fast. This architecture is light enough to run at real time on modern smart phones [19].

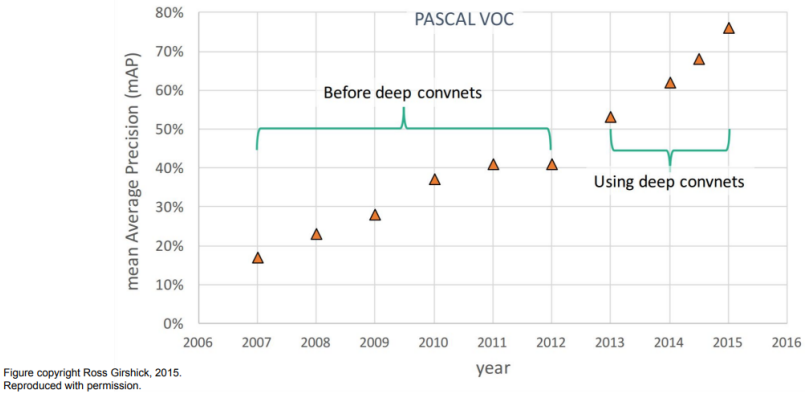

Figure 2.2: Mean average precission (mAP) of object detection before and after using deep learning techniques. (Li *et al.* [9], Slide 54).

Object detection algorithms also advanced using CNNs. The R-CNN architecture achieved a breakthrough by increasing the mAP "by more than 30% relative to the previous best result on VOC 2012" ([20], p.1) by extracting region proposals which are then classified. Reworking the architecture of the R-CNN model to increase the inference time, fast R-CNN [21] and faster R-CNN [22] have been developed. While faster R-CNN achieved 5 fps [22], the first algorithms allowing real-time object detection have been the YOLO [23] and the SSD [24] model. In YOLO, the image is divided into a set of grid cells and for each grid cell, a set of 5 bounding boxes are proposed. Each bounding box returns a score that reflects if the box contains an object and in addition to that a class prediction for each box. All of these bounding box informations plus its scores are fed into a Neural Network at once, instead of independently process the region proposals like in the R-CNN family, allowing YOLO object detection in real time [23].

Combing the problem of semantic segmentation and object detection leads to instance segmentation. The faster R-CNN structure has been extended to the mask R-CNN approach [25]. The mask R-CNN model computes the bounding box coordinates, the class prediction and a binary segmentation mask in three different output branches [8]. The mask R-CNN model was able to extend the faster R-CNN architecture by adding a FCN branch while still running at 5 fps [25]. Real time instance segmentation was achieved by by YOLACT++. YOLACT++ breaks up instance segmentation into two simpler tasks that can be performed in parallel. First, "prototype masks" are generated, second a set of "linear combination coefficients per instance" are predicted. Lastly, the prototypes are combined in a linear manner

using the predicted coefficients for each instance. This model is able to achieve real time (>30 fps) performance on only one powerful GPU [26].

2.2 Challenges

There are several problems that have to be considered during the project:

1. achieving a good tradeoff between high-quality results and low inference time (real-time performance)
2. achieving invariance towards environmental disturbances (varying light conditions)
3. working with high resolution images (4K)

The ICNet and Deeplab V3+ has already shown that 1. and 3. are possible to solve by using information form low resolution input images. 2. is highly dependent on the training data. Hence, it has to be made sure that the training data contains possible environmental changes that might occur in real world scenarios.

3 Method

This thesis will investigate the current state of the art semantic segmentation concepts and evaluate which model is best suited, given the task description. To achieve a high accuracy, training data will be generated. Since the main task is to separate humans (and objects they might carry), training data will be generated using a green screen. A green screen setup allows us to separate the background in a high quality manner using an alpha matting algorithm (e. g. [1]). The background in the training data can be virtually replaced with backgrounds that will be used in the photo booth in the final product. This results in high quality training data. It is very likely, that the model will not be able to generalize very well beyond the given backgrounds, since the training data environment is very much controlled and fixed. However, the final model does not have to generalize beyond the given scenario anyway.

Given the architectures in Section 2.1, one can see that they share similar concepts (Encoder-Decoder architecture, Region proposals, etc.). The project will most likely use either the YOLACT++ model, which is open source, delivers already high quality segmentation results and is able to run in real time or will adopt the Deeplab V3+ architecture, since it has already been shown that its complexity can be reduced heavily [19].

4 Preliminary structure

- Declaration of Authorship
- Abstract
- Contents
- List of figures
- List of algorithms
- Introduction
  - Motivation
  - Goal of the thesis
- Methods
  - Model
  - Training data
  - Preprocessing
  - Network training
- Results and discussion
- Evalutation
- Conclusion
- Acknowledgements
- Bibliography

5 Time frame

Week	**Date**	**What will be done**
1-2	01.04.2020 - 14.04.2020	fixing the topics that will be investigated, fixing the model that will be used
2-4	15.04.2020 - 28.04.2020	preparing and recording training data, preprocess training data
4-8	29.04.2020 - 26.05.2020	training the model, evaluating and analysing the results, start writing
8-12	27.05.2020 - 22.06.2020	fix potential issues, finish implementation, finish writing,
12	23.06.2020	finish thesis

6 Bibliography

1. Gastal, E. S. & Oliveira, M. M. Shared sampling for real-time alpha matting. *Computer Graphics Forum* **29,** 575–584. doi:10.1111/j.1467-8659.2009.01627.x (2010).
2. Sych, T., Brent, A., Phil, M. & Microsoft. *depth-camera @ docs.microsoft.com* 2019.
3. COCO Consortium. *COCO - Common Objects in Context* 2016.
4. Shahabeddin Nabavi, S., Rochan, M., Yang & Wang. Future Semantic Segmentation with Convolutional LSTM. *British Machine Vision Conference 2018, BMVC 2018,* 1–12 (2018).
5. Pfeuffer, A., Schulz, K. & Dietmayer, K. *Semantic Segmentation of Video Sequences with Convolutional LSTMs* in (2019). doi:10.1109/IVS.2019.8813852.
6. Shi, X., Chen, Z. & Wang, H. Convolutional LSTM Network. *Nips,* 2–3. doi:[] (2015).
7. Szeliski, R. *Computer Vision: Algorithms and Applications* 185–186. doi:10.1017/cbo9780511974076.010 (Springer, 2011).
8. Mittal, M., Arora, M., Pandey, T. & Goyal, L. M. Image Segmentation Using Deep Learning : A Survey, 41–63. doi:10.1007/978-981-15-1100-4_3 (2020).
9. Li, F.-F., Johnson, J. & Yeung, S. *Lecture 11: Detection and Segmentation* 2017.
10. Nobuyuki, O. A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics* **9,** 62–66. doi:10.1109/TSMC.1979.4310076 (1979).
11. Dhanachandra, N., Manglem, K. & Chanu, Y. J. Image Segmentation Using K -means Clustering Algorithm and Subtractive Clustering Algorithm. *Procedia Computer Science* **54,** 764–771. doi:10.1016/j.procs.2015.06.090 (2015).

12. Fukushima, K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics* **36,** 193–202. doi:10.1007/BF00344251 (1980).

13. Long, J., Shelhamer, E. & Darrell, T. Fully Convolutional Networks for Semantic Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **39,** 640–651. doi:10.1109/TPAMI.2016.2572683 (2014).

14. Noh, H., Hong, S. & Han, B. Learning Deconvolution Network for Semantic Segmentation. *Proceedings of the IEEE International Conference on Computer Vision* **2015 Inter,** 1520–1528. doi:10.1109/ICCV.2015.178 (2015).

15. Badrinarayanan, V., Kendall, A. & Cipolla, R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **39,** 2481–2495. doi:10.1109/TPAMI.2016.2644615 (2015).

16. Zhao, H., Qi, X., Shen, X., Shi, J. & Jia, J. ICNet for Real-Time Semantic Segmentation on High-Resolution Images. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **11207 LNCS,** 418–434. doi:10.1007/978-3-030-01219-9_25 (2017).

17. Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K. & Yuille, A. L. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **40,** 834–848. doi:10.1109/TPAMI.2017.2699184 (2016).

18. Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F. & Adam, H. in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 833–851 (2018). doi:10.1007/978-3-030-01234-2_49.

19. Bazarevsky, V. & Tkachenka, A. *Mobile Real-time Video Segmentation* 2018.

20. Girshick, R., Donahue, J., Darrell, T. & Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition,* 580–587. doi:10.1109/CVPR.2014.81 (2014).

21. Girshick, R. *Fast R-CNN* in *2015 IEEE International Conference on Computer Vision (ICCV)* **2015 Inter** (IEEE, 2015), 1440–1448. doi:10.1109/ICCV.2015.169.

22. Ren, S., He, K., Girshick, R. & Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE transactions on pattern analysis and machine intelligence* **39,** 1137–1149. doi:10.1109/TPAMI.2016.2577031 (2015).

23. Redmon, J., Divvala, S., Girshick, R. & Farhadi, A. *You Only Look Once: Unified, Real-Time Object Detection* in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* **2016-Decem** (IEEE, 2016), 779–788. doi:10.1109/CVPR.2016.91.

24. Liu, W. *et al.* in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 21–37 (2016). doi:10.1007/978-3-319-46448-0_2.

25. He, K., Gkioxari, G., Dollár, P. & Girshick, R. Mask R-CNN. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **42,** 386–397. doi:10.1109/TPAMI.2018.2844175 (2017).

26. Bolya, D., Zhou, C., Xiao, F. & Lee, Y. J. YOLACT++: Better Real-time Instance Segmentation, 1–12 (2019).