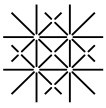


University of Basel

<A TITLE>

Bachelor-Thesis

Natural Science Faculty of the University of Basel
Department of Mathematics and Computer Science
Computer Graphics and Vision (Gravis)
http://gravis.dmi.unibas.ch/

Examiner: Prof. Dr. Thomas Vetter
Supervisor: Dr. Adam Kortylewski

Elias Arnold
elias.arnold@stud.unibas.ch
14-930-770

10.08.2018

Acknowledgments

I would like to thank Dr. Adam Kortylewski as my main supervisor for the guidance and insight through the work and writing of this thesis. In addition, I would like to thank Prof. Dr. Vetter for the possibility to write this thesis in his research group.

Abstract

This thesis evaluates a pretrained Fully Convolutional Network to segment images into face and non-face region. The result of this segmentation is a binary encoded image. Such a segmentation can be used to limit an algorithm to a specific image region, so that not the entire image has to be scanned by the algorithm.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Table of Contents

1 Introduction

Fitting is the process of generating a 3D model of a face from a 2D image. There are different approaches on how to do that. The gravis group of the University of Basel has developed an MCMC (Markov Chain Monte Carlo) algorithm that makes random changes to a 3DMM (3D Morphable Model) and accepts the proposal face whenever the new 3D face has a greater probability than the previous one. For this Thesis, we use the popular Basel Face Model 2017 [1]. That the algorithm can determine the likelihood of a 3D facial proposal relative to a given 2D face, the algoritm's evaluator needs to now, which pixels to include in the probability calculation. Therefore we have to label each pixel if it's part of the face which should be regarded by the evaluator, or if it's background and hsn't any importance for the construction of the 3D Model. Nirkin et al [2] claim, that this is possible with a standart fully convolutional network.

The idea of Artificial Neural Networks was heavily influenced by biology. They consist of a variety of neurons which are grouped in layers. The way each neuron works is very simple. It takes multiple inputs of varying strength from other neurons, sums them up and decides depending on the sum whether it should send a stimulus itself and if so, in which strength. Each layer is somehow connected to the next layer. Some layers are fully connected (each neuron of a layer is connected to every other neuron in the next layer) while others are convolutional. Convolutional means that a neuron only gets input of its neighbors in a previous layer. There are many different architectures which mainly differ in the number of layers, number of neurons per layer and the interconnectivity of the neurons. A classical Convolutional Neural Network (CNN) is depicted in 1.4.

Already in 1943 Warren McCulloch and Walter Pitts [3] showed that even simple networks of this kind can simulate every possible logical formula. For this they used a neuron model that consisted of simple logic gates and could process only binary input and output signals.

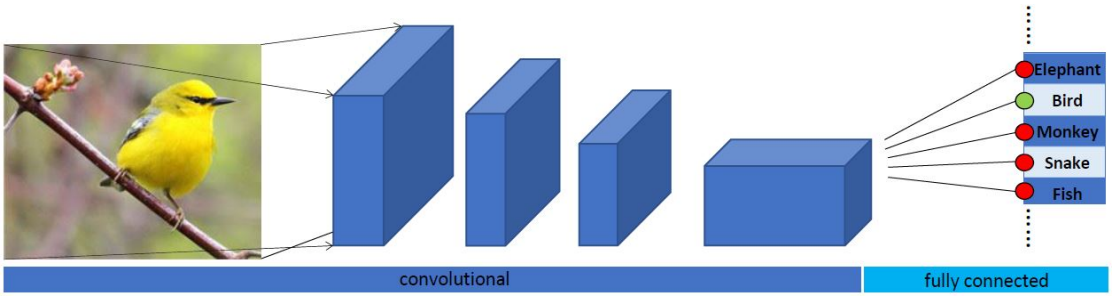


Figure 1.1: In the lower-left corner, you see the 16 layers of the VGG16-FCN. Each Max-Pooling layer cuts the input image size in half. In the top-right, you can see the meaning of the '8s' term of the FCN-Name. It means that the resulting image has to be 8x upsampled, to get an image which is in size equal to the input image (which means we have four pooling-layers)

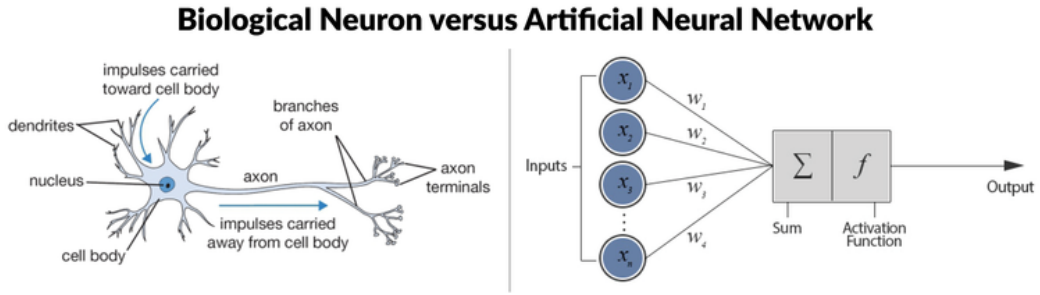


Figure 1.2: The left-hand side of the image [1]shows a Biological Neuron. It is a nerve cell that occurs in almost every animal. On the right-hand side is On the right is an artificial neuron. It sums up the stimuli of the previous neurons, applies an activation function to this sum and forwards the output of this function itself

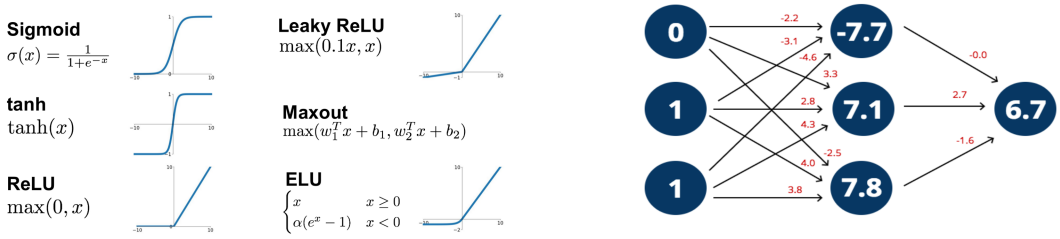


Figure 1.3: TODO!

[1] Source: Google.com

1.1 The network used

For this thesis, we used a pretrained fully convolutional network from [2]. A Fully Convolutional Network (often called: FCN) is basically a CNN but with a modified architecture. An FCN doesn't have the fully connected layers usually found at the end of an CNN. These layers would enable the Network to make decisions based on global information. Aa CNN can for example be used for classification. But for image analysis we want local information of the input image (we don't want to know if there is a face in the image, but where the face is in the image). Therefore a FCN uses only convolutional and pooling layers. In the whole Fully Convolutional Network only the following structure is repeated: One or more convolution layers and a pooling layer which downsamples the picture. This constellation is repeated several times.

The assembly of our network used for this thesis follows the FCN-8s-VGG architecture with extensions of [4]. The first part 'FCN' stands for 'fully convolutional network', '8s' means that the result gets eight times upsampled (4 pooling layers) and 'VGG' means the popular 16-layer network used by Oxford's Visual Geometry Group [5]. The original task of the network was to find the name of an object in an input image. The network could distinguish between 1000 different objects. Each cell in the final Softmax-Layer (1*1000 in size) was a boolean variable for one specific object.

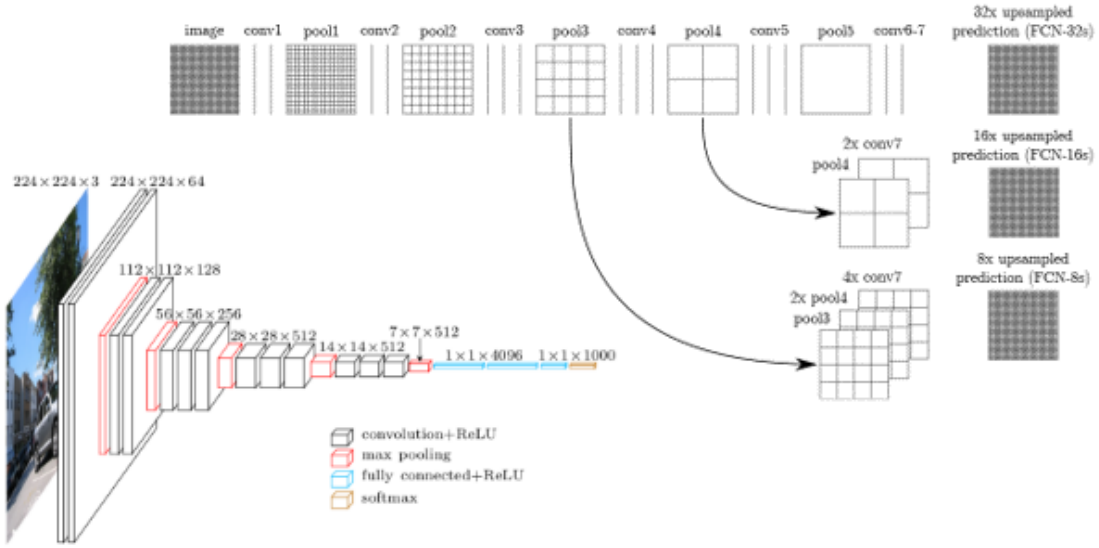


Figure 1.4: In the lower-left corner, you see the 16 layers of the VGG16-FCN. Each Max-Pooling layer cuts the input image size in half. In the top-right, you can see the meaning of the '8s' term of the FCN-Name. It means that the resulting image has to be 8x upsampled, to get an image which is in size equal to the input image (which means we have four pooling-layers)

1.2 Related Work

For our experiments we used a pretrained Fully Convolutional Network (FCN) of [2]. They showed that even with a widespread network, you can do good segmentation and that the network does not have to be specially tailored to the future purpose. But the network must have been trained with a large enough data set. For more details about the used FCN and the architecture see section 1.1. They used the FCN for intra- and intersubject face swapping on the Labeled Faces in the Wild (LFW) data set and showed that intra-subject swapped faces remain as recognizable as before the swap and that in the intersubject version better face swapping leads to less perceptibility.

They used a semi-supervised approach to produce training data in order to train the FCN. To produce large quantities of them, they used 2'043 face videos of the IARPA Janus CS2 dataset. To avoid searching for the face in every frame of the video they used motion queues which tracked the face based on an initial segmentation based on [6] which enriched their trainingset to 9'818 images. To enlarge the collection of images, they rendered 3D Shapes of of various objects (e.g. sunglasses, hands) into existing images. Each occlusion added 9'500 images to their trainingset.

The gravis group of the University of Basel developped "Occlusion Aware 3D Morphable Models" [7]. These Methods use an iterative approach to generate the z-labels, namely to label each pixel wether it belogs to the face or is background. This approach can handle multiple labels and differentiate between multiple occlusion types. For example face specific ones (eg. beards, sunglasses) and background. For updating the z-labels they use an algorithm which classifies a pixel based on the probabilities for each possible label. But for our experiments we limited ourselves to two. We only need to distinguish face (including skin and beard) and background.

The algorithm to update the z-labels as mentioned before is an EM-algorithm like method to solve two problems simultaneusly. In the E-steps they updated the z-labels and in the M-steps they updated the face parameters. Conventional approaches often fail on important parts of the face such as the eyes, eyebrows or the oral region because oftheyr strong variability in color and shape. The segmentation of [7] also has difficulties with these aspects as Figure 1.5 shows. The algorithm starts with an initial guess and then altenating updates the Parameters Θ and the z-labels. From the updated Parameter Set (M-Step) the algorithm updates the z-labels (E-Step) and vice versa (see Figure 1.6).

An approach using convolutional neural networks to segment occluded faces has already been described by [9]. The big difference to our approach is that multiple frames are needed for the

[2] Figure 1.6 and it's description are one to one coped from Fig.4 of the following paper [7]

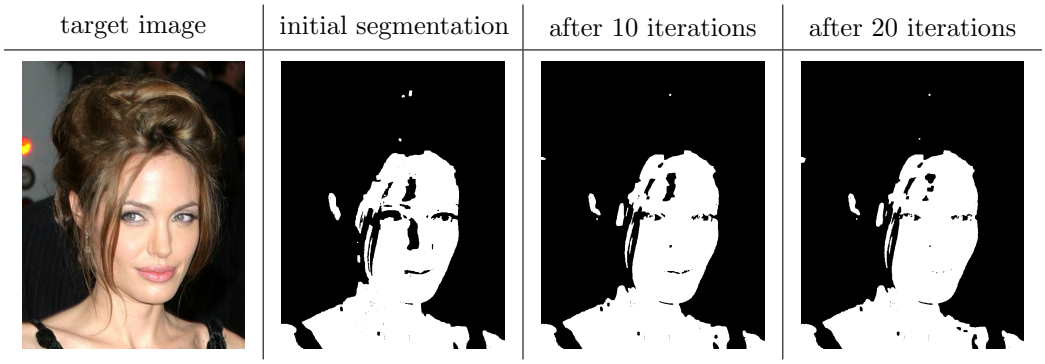


Figure 1.5: It's obvious that the initial segmentation doesn't include the whole face region. Striking in this sample image are not only the eyes as mentioned before but also the shadow of the nose, which is first segmented as a background. Only after a certain number of iterations these errors are partially recognized and provided with the correct label.

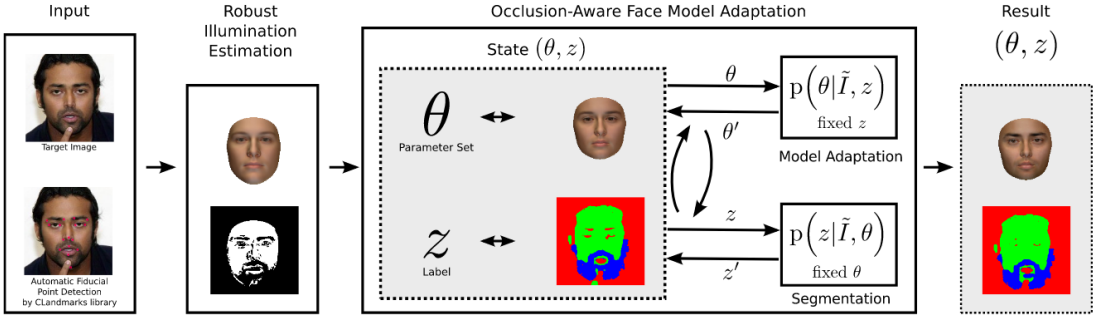


Figure 1.6: [2]Algorithm overview: As input, we need a target image and fiducial points. We use the external Clandmark Library for automated fiducial point detection from still images (Uřičář et al (2015) [8]). We start with an initial face model fit of our average face with a pose estimation. Then we perform a RANSAC-like robust illumination estimation for initialization of the segmentation label z and the illumination setting (for more details see Figure 9). Then our face model and the segmentation are simultaneously adapted to the target image I. The result is a set of face model parameters Θ and a segmentation into face and non-face regions. The presented target image is from the LFW face database (Huang et al. (2007))

final segmentation. An other approach of [10] integrate occlusion into the fitting process to model an uncertainty. They use random forests to detect facial-occlusions by hair.

- Work of Nirkin et al. (Their results)
- Summarize Paper: "Occlusion-aware 3D Morphable Models and..."
- Work of Saito et al. Work of Morel-Forster (2017)

1.3 Expectations of the FCN

Because [2] trained the FCN on a very large and diverse dataset, we expect the network to perform well in the task of segmenting a face. They claim that a standart segmentation network is sufficient to segment as well as a network especially tailored for this task and outperforms state of the art methods for face segmentation. We create multiple synthetic faces and render various occlusions over them. We expect the FCN to do well on usual facial-occlusions (eg. hands, microphones).

- We expect the FCN to perform well of occlusions used for training, e.g. hands, microphones
- An FCN is known to be very fast, so we expect to speed up the fitting.

2
Evaluation

2.1 Evaluation on Datasets

The face segmentation network was tested on two different datasets 1.) The Caltech occluded faces in the wild (COFW) and 2.) parts-labeled LFW Dataset of the University of Massachusetts.

Nirkin et al. [2] claim that both the face itself an the context of the face play an important role for the outcome of the segmentation. To measure this effect and reduce the impact of outliers, we repeat each experiment multiple times with a different face and a different backgroundimage. Within a dataset The face and the backgroundimage stay the same. For the results we use the average of all results.

2.1.1 COFW

Since on the COFW dataset, only landmarks and bounding-boxes are given, the segmentation had to be evaluated qualitatively. We weren't able to count the correctly labeled pixels of the with respect to a ground-truth mask. To increase the precision of the network, we cropped the images according to the provided bounding-boxes. Because the bounding-boxes were only inluding the eyes and the mouth of the subject, we had to add an offset which was optically determined. Therefore, we measured the quality of the neural network only qualitatively. Nevertheless, we tried to reconstruct the graphic on Nirkis[1] github-page.

Figure 2.1: 18 images of the COFW Dataset overlayd with the FCN output (in red). The given box was extended upwards by 80%, to the right and left it was 55% and downwards t was 5%.

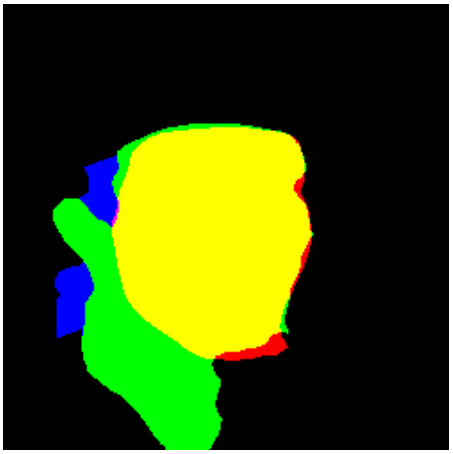
2.1.2 Parts-LFW

In the Parts-LFW dataset, we had a ground-truth mask for every image. The make distinguishes between hair, skin and background. We iterated trough all the provided groud-truth masks and overlaid them with the segmentation of the FCN. The evaluation is quite impressive. The FCN performs very good in segmenting only pixels which belong to the face. In average over the 500 images of the Parts LFW dataset, there are 98.5% right non-segmentations. On the other hand, only 85.4% of all the pixels which belong to the face are segmented as face pixels.Figure (TODO!) depicts such a mask. Unfortunately, on the given mask, no distinction is made between face and other parts of the body, but everything is segmented as skin (Subfigures 2.2(a) and 2.2(b)). However, the bigger problem is that some faces have beards. The FCN of Nirkin[1] segmented facial hair which was excluded on the provided labels. So we had to manually remove these images (Subfigures 2.2(c) and 2.2(d)). To reduce the effort, we took the Parts-LFT Validation set containing 500 images. After removing the ones with a beard or mustache, we were left with 447 images. The results are summarized in the following table

2.2 Evaluation on synthetic-data

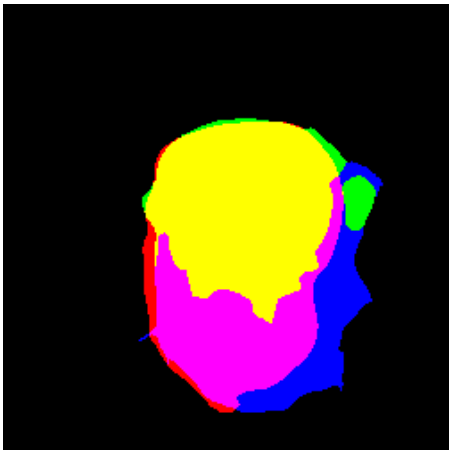
In order to evaluate the FCN on synthetic-data, we used the Parametric-Face-Image-Generator of [11] to produce images of a random face in a given pose. We extended the software so that it now renders occlusions over the face. Further, the Parametric-Face-Image-Generator now produces a Ground-Truth-Mask, which classifies every pixel either as part of the face or as non-face.

(a) An facial-image with the segmentation of the FCN highlited in red



(b) The provided ground-truth segmentation of the image 2.2(a) overlaid with the mask of the FCN (red)

(c) An facial-image with the segmentation of the FCN highlited in red



(d) The provided ground-truth segmentation of the image 2.2(c) overlaid with the mask of the FCN (red)

Figure 2.2: Plots of four Turing machines

false-positives (hair)	7.04%
false-positives (background)	0.68%
right-segmentations	85.87%
right non-sementations	99.32%

Figure 2.3: TODO!

2.2.1 Dpendence of the Euler angles

Because we can now create synthetic face images in any desired pose, we first wanted to find out the accuracy of segmenting the FCN for the angles: Yaw, Roll, and Pitch. In order to do

that, we produced with the tool of [11] 101 face images for every angle from $-50°$to $50°$. In every picture, the face is turned one degree further. We evaluated each angle itself and every possible combination of the angles in order to to create a hierarchy under the angles:

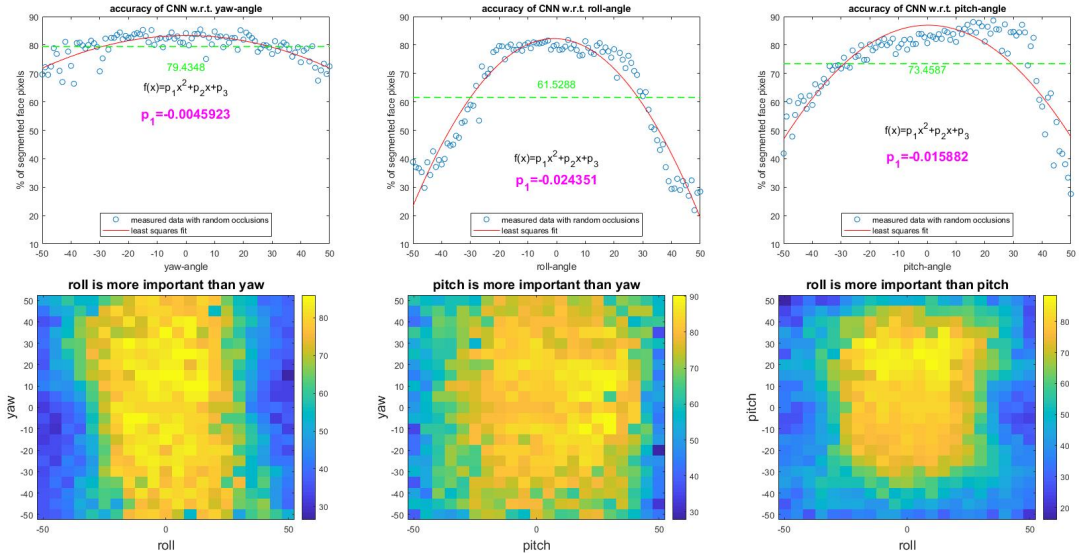


Figure 2.4: In the plots on the top row we see the segmetation accuracy in percent (on the y-axis) for every single image (with face angle from $-50°$to $50°$on the x-axis). The point cloud is approximated by a quadratic function via a least squares fit (red curve / $f(x)$). The first parameter of this function determines the opening angle (p_1). From these graphs we can conclude that the roll angle is the most relevant for the FCN. The pitch-angle is less important and that the accuracy of the FCN is still good even with hith angles (least important). In the bottom row the colors indicate the segmentation accuracy. The brighter the color, the better the segmentation. In every plot, there is a cluster of high accuracy segmentations centered in the origin. The angle on whose axis the cluster has the lesser extent is the more important of the two. We call an angle important, when a small change of this angle leads to a failure of the FCN.

2.2.2 random boxes as occlusions

With the parametric-face-image-generator of [11], we produced about 5.2 thousand images of Which for every angle from $-40°$to $40°$include 100 images of 5 different faces with 20 different occlusion levels. In the given table (Figure 2.5) all provided images show a face, from which 20% of the pixels are occluded by a randomly colored box. We can optically verify, that 1) the yaw angle, despite the occluding box, hasn't much of an effect. 2.) In both situations, -40 and 40, of the roll angle, the result is not satisfying. 3.) In the third column, the segmentation with a negative pitch angle is much worse than with a positive one! Note that the results for each grid cell in figure 2.6 is based on 5 segmentations (5200 images over all)!

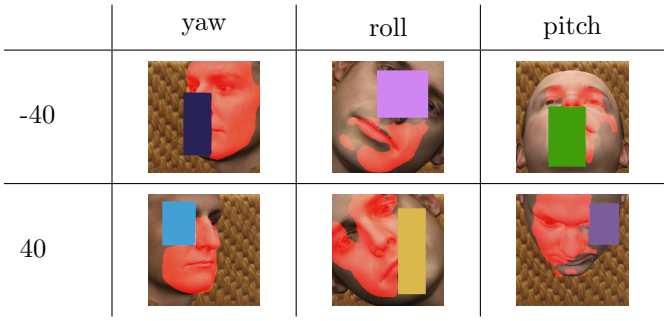


Figure 2.5: asdfasdfTODO!

Although boxes as occlusions are very simpley and we're in control of the rectangle's size, we can occlude a given amount of the face region with this method, but in practice, exact rectangles are very rare. That's why we left it with the rectangles and used real-world objects (e.g. hands, microphones or sunglasses) as synthetic occlusions. Unlike Nirkin[1], we didn't use the landmarks of the face for this task. We placed them randomly on the image instead. Thats the reason why we use a second plot to show the percentage of occluded face pixels in the following.

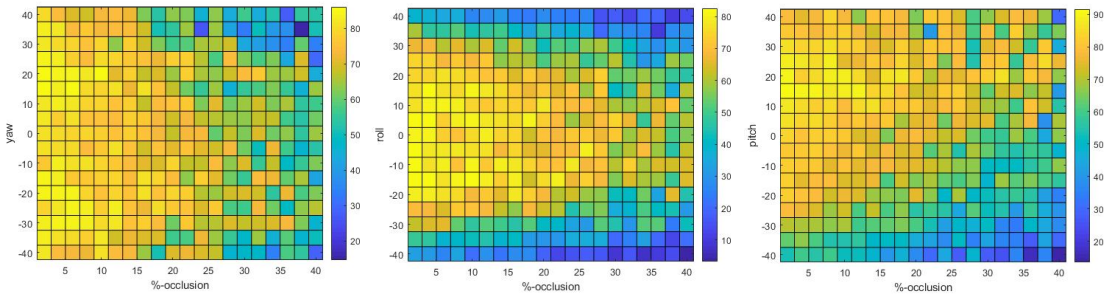


Figure 2.6: The color of each grid cell indicates the accuracy of the segmentation of the FCN on a set of faces turned by the corresponding angle and occluded with a square, so that the corresponding amount of the face is masked. The brighter the color, the better the segmentation. On each plot, we would expect to see a triangle pointing to the right. This means that the combination of a large angle and a large occlusion make the face even more unsegmentable.

We see that the segmentation is very sensitive to roll-angles and the FCN is not trained to segment faces in every rotation! Surprisingly, the yaw angle plays a subordinate role here. The right most plot tells us, that the sign of the pitch angle plays a significant role! Since we aren't able to determine at which angles exactly the FCN begins to fail,we repeated the experiment with a higher angle range:

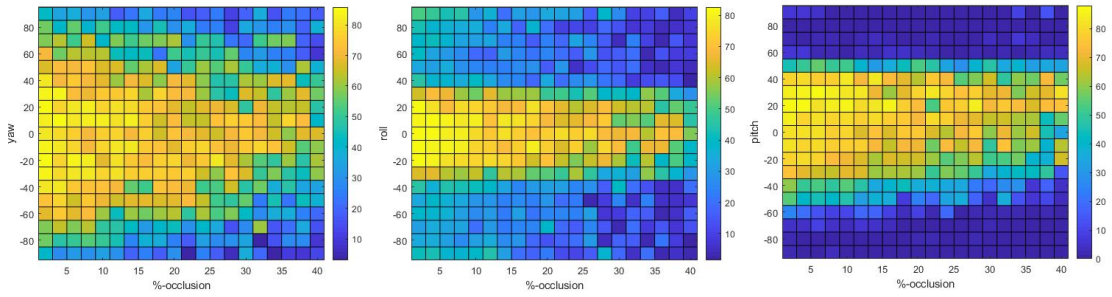


Figure 2.7: This plot shows the outcome of a similar expreiment as snown in Figure 2.6 with less resolution. All the angles (yaw, pitch and roll) range from -90°to 90 °. The scale on the "%-occlusion" stays the same as in Figure 2.6 We can clearly see the limits of the FCN even with a occlusion of 2%! Very interestig is the hard transition from good segmentations to bad segmentations in the two right plots.

3

Comparison with other approaches

- setup
- compare the Fitting Results depending on the mask used
- GROTRU vs. EGGER vs. FCN vs. DUMMY

3.0.1 setup

3.0.2 quality

3.0.3 time

4

Integration of the FCN into the original work of Egger et al.

5
Conclusion

- What can the FCN do?
- What are the advantages towards the Method of Egger

Bibliography

[1] Gerig, T., Forster, A., Blumer, C., Egger, B., Lüthi, M., Schönborn, S., and Vetter, T. Morphable Face Models - An Open Framework. *CoRR*, abs/1709.08398 (2017). URL http://arxiv.org/abs/1709.08398.

[2] Nirkin, Y., Masi, I., Tran, A. T., Hassner, T., Medioni, and Medioni, G. On Face Segmentation, Face Swapping, and Face Perception. In *IEEE Conference on Automatic Face and Gesture Recognition* (2018).

[3] McCulloch, W. and Pitts, W. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biology*, 5(4):115–133 (1943).

[4] J. Long, E. S. and Darrell, T. Fully convolutional networks for semantic segmentation. In *Proc. Conf. Comput. Vision Pattern Recognition*, page 3431–3440 (2018).

[5] K. Simonyan, A. Z. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *arXiv:1409.1556*.

[6] M. Grundmann, M. H., V. Kwatra and Essa, I. Efficient hierarchical graph-based video segmentation. In *Proc. Conf. Comput. Vision Pattern Recognition* (2010).

[7] Bernhard Egger, A. S. A. K. A. M.-F. C. B. T. V., Sandro Schönborn. Occlusion-aware 3D Morphable Models and an Illumination Prior for Face Image Analysis (2018).

[8] Uricar, M., Franc, V., Thomas, D., Sugimoto, A., and Hlavac, V. Real-time multi-view facial landmark detector learned by the structured output SVM. In *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, volume 02, pages 1–8 (2015).

[9] Saito, S., Li, T., and Li, H. Real-Time Facial Segmentation and Performance Capture from RGB Input. *CoRR*, abs/1604.02647 (2016). URL http://arxiv.org/abs/1604.02647.

[10] Morel-Forster, A. *Generative shape and image analysis by combining Gaussian processes and MCMC sampling*. Ph.D. thesis, University of Basel, Faculty of Science (2016).

[11] B. Egger, A. M.-F. A. S., A. Kortylewski. parametric-face-image-generator (2017).

A

Appendix

Declaration on Scientific Integrity
Erklärung zur wissenschaftlichen Redlichkeit

includes Declaration on Plagiarism and Fraud
beinhaltet Erklärung zu Plagiat und Betrug

Author — Autor
Elias Arnold

Matriculation number — Matrikelnummer
14-930-770

Title of work — Titel der Arbeit
<A TITLE>

Type of work — Typ der Arbeit
Bachelor-Thesis

Declaration — Erklärung
I hereby declare that this submission is my own work and that I have fully acknowledged the assistance received in completing this work and that it contains no material that has not been formally acknowledged. I have mentioned all source materials used and have cited these in accordance with recognised scientific rules.

Hiermit erkläre ich, dass mir bei der Abfassung dieser Arbeit nur die darin angegebene Hilfe zuteil wurde und dass ich sie nur mit den in der Arbeit angegebenen Hilfsmitteln verfasst habe. Ich habe sämtliche verwendeten Quellen erwähnt und gemäss anerkannten wissenschaftlichen Regeln zitiert.

Basel, 10.08.2018

Signature — Unterschrift