Contents

Automatic Segmentation of Zebrafish using Deep Learning

Seema Negi

1 Introduction

The task of the project is to segment the Zebrafish and quantify its few morphological features. The zebrafish is a well-established model organism used in biology and medicine. Due to its small size, the optical transparency of complex organs, and ease of culture, the zebrafish larva is an ideal organism for large-scale screening. Zebra Fish is considered an important model in understanding diseases such as cancer. Researchers are daily collecting images and looking at how different diseases and medication are affecting the zebrafish. Accurate segmentation and morphological features can help in evaluating size changes in zebrafish and can unlock many aspects that can lead to a major change in biology. There are various solutions available where different model-based and deep learning techniques like FCN, Hybrid Models, GAC (3) etc are used for zebrafish segmentation. For our project, we have used deep learning but with a different layer architecture (SegNet). We got data from João Campos Costa at the Genome Engineering ZebrafishSciLifeLab facility at Uppsala University. The data consist of 6 different folders which contain images of raw data and ground truth (Segmented). The name on each folder describes the age (dpf – days post fertilization) and the number following the age indicates that they are different types of fish. The fish images are RGB images of 250x1024 size while the segmented images are binary images with 250x1024 size.

2 Method

2.1 Zebrafish Segmentation

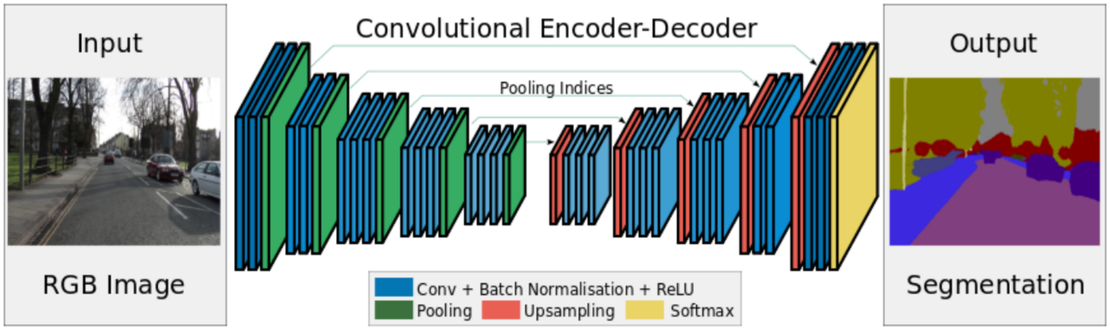

Figure 1: Segnet Architecture

For this task, semantic segmentation is performed using deep learning. A semantic segmentation network classifies every pixel in an image, resulting in an image that is segmented by class. We have used SegNet which provides a deep convolutional encoder-decoder architecture for pixelwise image segmentation. This architecture was first researched and developed by members of the Computer Vision and Robotics Group at the University of Cambridge, UK.(2)

SegNet uses a sequence of non-linear processing layers known as encoders and a set of layers as decoders and these are followed by a classifier which is a pixel based classifier. There are one or more convolutional layers with batch normalisation and a ReLu in each encoder and further, these are followed by non-overlapping maxpooling and sub-sampling. In the decoding layer pooling process is un sampled with the help of max-pooling. Low resolution feature maps are un sampled in the decoders using max pooling indices. Hence high frequency details in the segmented regions are kept and reduction of parameters are done in this layering structure(2). The model was trained with the following hyperparameters :
Solver = sgdm(Stochastic gradient descent), Learning Rate = 0.01, MaxEpochs = 10, Momentum = 0.9, Shuffle = every-epoch

2.2 Quantification of morphological features

Area, length, centroid and minor axis length on the segmented results were computed using MATLAB's built-in functions. Centroid tells the information of x and y coordinates into a two columns matrix.

3 Implementation

Pseudocode for Automatic Zebrafish Segmentation using Segnet

1: Load training and corresponding ground truth images using imageDatastore()
2: **for** All Ground truth images **do**
3:     Convert the binary image to rgb image using
4:         $\mathbf{I}_{rgb} = 255 * repmat(uint8(I_{binary}), 1, 1, 3)$
5: **end for**
6: Define two classes as ”fish” and ”background”
7: Define label Ids as [255 255 255] and [0 0 0] for ”fish” and ”background” respectively
8: Use PixelLabelDatastore() to read pixel label data from ground truth images
9: Divide training data and pixel data in 60:20:20 of training, validation and testing set.
10: Create network layers using segnetLayers() with encoderDepth = 4
11: Define training options
12: Data augmentation using random left/right reflection and random X/Y translation of pixels
13: Combine training,augmented data using pixelLabelImageDatastore()
14: Train network with data collected in Step 13 , layers from Step 10 and training options defined in Step 11
15: Perform segmentation on an image using semanticseg(I, net)
16: **for** All images in test data set **do**
17:     Segment using results = semanticseg(testDataSet,net)
18:     Compute and Evaluate segmentation results using metrics = evaluateSemanticSegmentation(results,groundtruth);
19:     Get Data metrics and observe the results
20: **end for**

The above pseudo code was implemented in MATLAB.

4 Results

For the evaluation of our model, the dice score was computed to measure the spatial overlap between the segmented image and available ground truth image. For all the images of different dpf's, the dice coefficient similarity measure was more than 0.9.

The following evaluation metrics values were obtained on the test dataset:

Accuracy	Jaccard Coefficient	F1 score
0.97969	0.96024	0.97544

The figures 2 and 3 show the images of ground truth, segmented image from our model, outlined segmentation on original image and dice similarity measure using ground truth and segmented image of 2dpf and 3dpf zebrafish respectively.

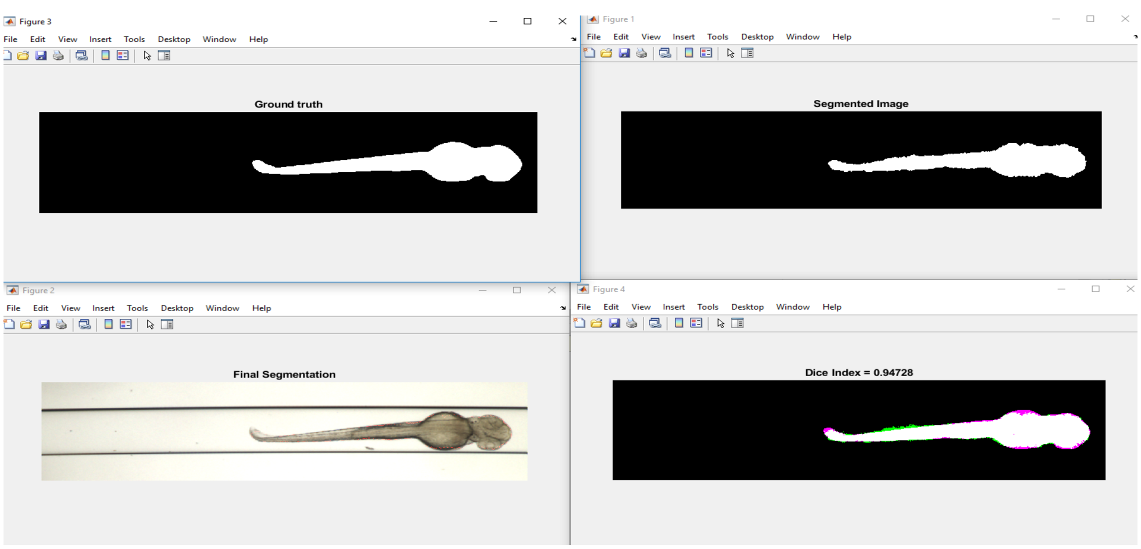

Figure 2: 2dpf Segmentation

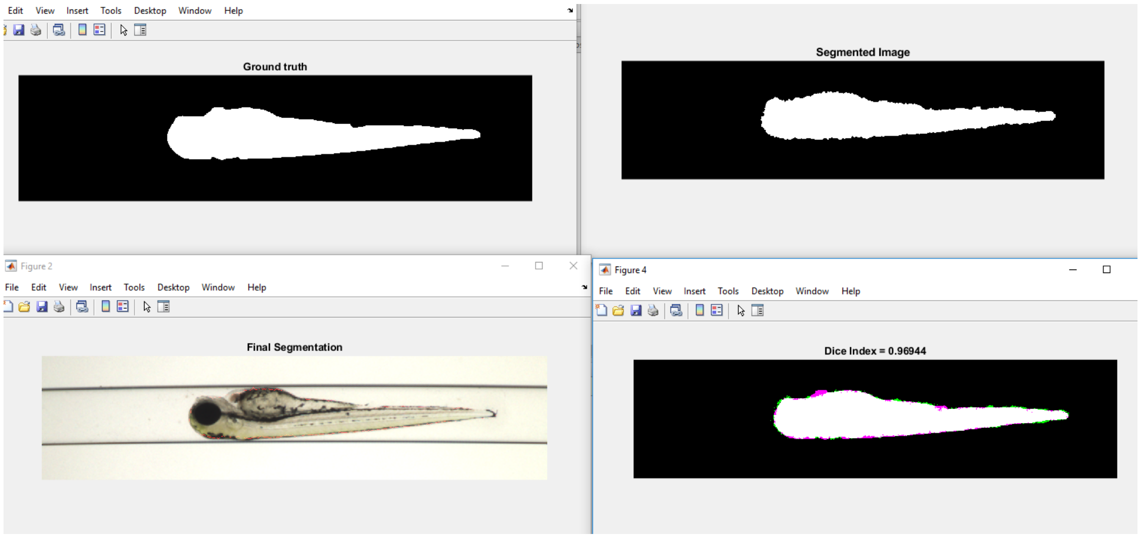

Figure 3: 3dpf Segmentation

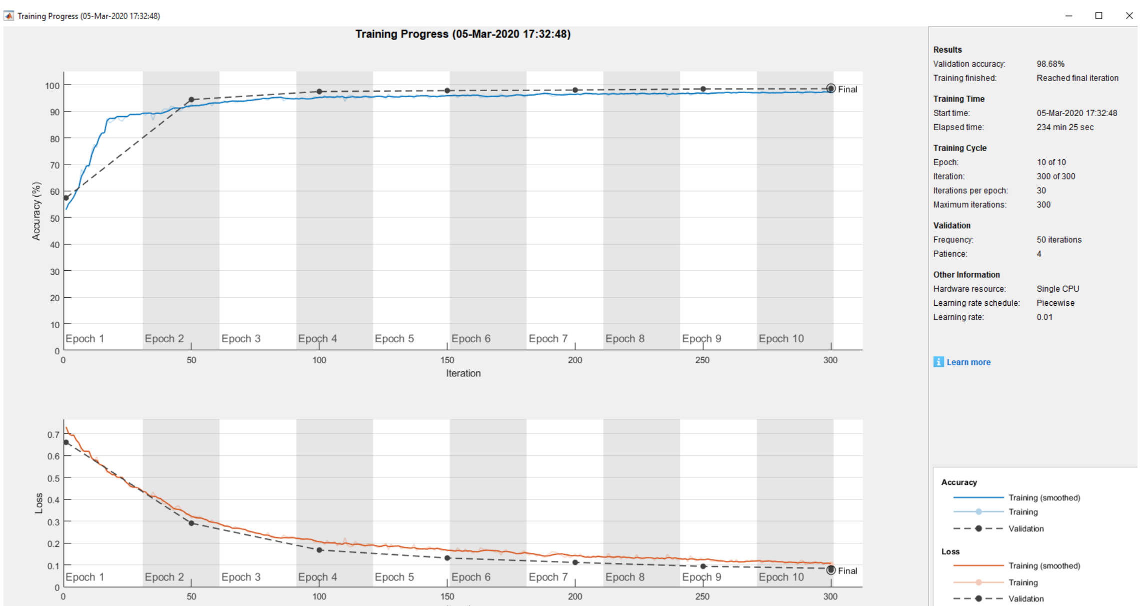

Figure 4: Training Results with 10 epochs

Figure 4 shows the accuracy and loss of the network with 10 epochs.

5 Discussion

Good segmentation results could be obtained with end to end semantic segmentation using deep learning. The model can learn essential patterns efficiently to provide expected results. Though there are few prerequisites and limitations also. Choosing the best network and tuning the hyperparameters could be a tedious task (4) while training a network could take large time and results in high computational cost. For our model, it took around 6-7 hours with only 10 epochs. Changing the parameters to observe how the model behaves and training it, again and again, is very time consuming. We tried automatic segmentation with FCN models also but GPU memory issue restricted the completion of the task. Also, for deep learning, we need large training data which is not the case with model-based segmentation. For our task, we tried some model-based solutions also. Mean shift algorithm was applied to the input image to obtain an overview shape description of the object. This improves the discrimination of the specimen to the background and provides a segmentation candidate retaining the overall shape of the zebrafish (3). The resulted image was converted to a binary image. To get better results we tried removing the capillary tube lines which were one of the difficult tasks as the lines were not straight, uniform and parallel. To conclude, a deep learning network when trained with enough data and best hyperparameters, can provide optimal results.

References

[1] https://se.mathworks.com/help/vision/examples/semantic-segmentation-using-deep-learning.html

[2] http://mi.eng.cam.ac.uk/projects/segnet/demo.phpresearch

[3] Yuanhao Guo1,Zhan Xiong1 ,Fons J. Verbeek1. An efficient and robust hybrid method for segmentation of zebrafish objects from bright-field microscope images

[4] https://www.hindawi.com/journals/cin/2019/8214975/