Project 4: Cilia Segmentation

Team Coburn• 04.10.2018

Vibodh Fenani
Layton Hayes
Zach Jones
Raj Sivakumar

Overview

Recent progress

- Minimum variance Threshold
- Ciliary Beat Frequency
- Optical Flow

Biggest risks

Tiramisu network

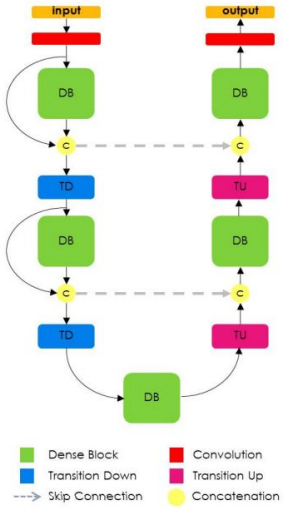
U-net

Variance Thresholding

Overview

- Compute pixel-wise intensity variance
- For each pixel if variance > threshold, mark as cilia
- threshold is tunable parameter
- Strange: found that constant threshold 9.05 performed better than a function of the variances
- Score: 0.23

Tiramisu (Fully Convolutional DenseNet)



Overview

- Implements downsampling-upsampling convolutional architecture but with DenseNets
- Main idea: get U-Net performance with fewer parameters

Troubles

- Difficulty training
- Trained network output nonsense
- Unsure if open-source implementation was correct

Ciliary Beat Frequency

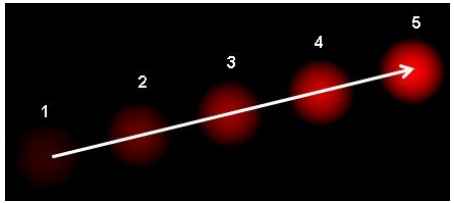
Pixelwise FFT

- Calculate FFT at each pixel across time
- Used numpy.fft after some experimentation
- Get mask of strongest frequency

Threshold

- Classify pixels that have the most energy at 10-12 Hz as cilia.
- Unfortunately not very successful.

Optical Flow



Preprocessing step

- Used the Gunner - Farneback algorithm from OpenCV
- Returns optical flow in x and y between each pair of images

Thresholding

- Calculated magnitude of flow
- Compared pixelwise mean and variance to a variously formulated thresholds to attempt segmentation
- Interesting but overall poor results

UNet

Preprocessing step

- Resized the images to Equal size
- Stored all the results in a numpy array

Threshold

- Converted all the pixels with >0.5 as cilia

Thanks!

Questions?