<Deckblatt>

# Abstract

Index

[Abstract 2](#_Toc494796379)

[1. Introduction 5](#_Toc494796380)

[1.1 Detection of neurodegenerative diseases 5](#_Toc494796381)

[1.2 Currently no early detection CSIRO and retinal image analysis 5](#_Toc494796382)

[1.3 Motivation 5](#_Toc494796383)

[2. Methods 5](#_Toc494796384)

[2.1 Quality preprocessing 5](#_Toc494796385)

[2.2 Stabilize movement of the eye 6](#_Toc494796386)

[2.3 Single Image Width Measurement 7](#_Toc494796387)

[2.3.1 Optic disc detection 7](#_Toc494796388)

[2.4 Extracting parameters from a Sequence of images 8](#_Toc494796389)

[2.5 Post processing of parameters 8](#_Toc494796390)

[4. Results 8](#_Toc494796391)

Abbildungs- Tabellenverzeichnis

# Introduction

## Detection of neurodegenerative diseases

Late onset Alzheimers Disease: PET & MRT

## Currently no early detection CSIRO and retinal image analysis

Vessel properties (width), correlation with diseases

Previously applied only on single images, expand into sequence

## Motivation

Develop new diagnosis method

Research vessel properties, measure vessel elasticity in IR sequence

# Methods

For the first time ever, a procedure had to be developed at the eHealth research center of the CSIRO in order to collect and evaluate information on the back of the eye, which was intended to be applied to a sequence of retinal images. The aim was to investigate whether the change in vessel width caused by the pulse was optically measurable. The basis for measuring the width of vessels in a single image was developed by Phd. Shaung Yu, so that this solution could be integrated.

The medium that is evaluated is a monochromatic infrared image of the human retina with a resolution of 1600x1200 pixels. To process a sequence of such images, a pipeline with three main steps and many sub steps has been developed. It is used to extract many different parameters from the image sequence, one of them being the change in vessel thickness over time. In addition, other pipeline-relevant methods are also implemented, which are mandatory for their regular flow as well as very helpful for the evaluation and verification of the intermediate step results.

In the following paragraphs, the most important methods of the pipeline for evaluating and analyzing vessel width are explained.

## Quality preprocessing

The individual frames of the image series sometimes have strong differences in quality. These quality differences range from small contaminants or particles that hover over a certain area during the series and cover some vessels, up to large distortions of the entire image through rapid movement and blinking. Such frames must be identified and excluded beforehand so that the analysis can be applied to a series of images that are as undistorted and uncovered as possible. Therefore, it is very difficult to develop an automated method because of the different nature of the distortions that can occur. The estimation of when a frame of a series should be excluded must, therefore, be done manually and at one's own judgment. To do this, several frames are taken into account before and after a blinking or abrupt movement of the eye and identified where distortion such as stretching of the whole or part of the image can be seen. This step is of highest priority, as distortion changes the vessel width enormously and can ultimately lead to completely wrong measurement values. Removing individual frames from a series has no effect on the following pipeline except that fewer records are available and everything related to it. This could possibly cause inaccurate readings, as important events such as the systolic peak of a vessel could be overlooked. However, given the consequences of distortion and the fact that affected frames cannot be analyzed anyway, the influence of remote frames is neglected.

## Stabilize movement of the eye

If the quality of the image series is as undistorted as possible, it must be stabilized, because for an exact measurement of the parameters the vessels should be at a fixed position throughout the sequence. The reference is the optical disc, which can be easily recognized as the origin of the retinal vessels and by the dark circle. There is always a slight movement of the retina in every series of pictures. However, a wide range of movements is usually observed, be it simple movements such as the slow, continuous drifting of the retina from one point to another, or complicated movement patterns such as sudden twitching of the eye from one point to another in different directions. The latter can also result from the removal of individual frames from the quality selection, so that due to missing frames the retina suddenly appears at a different location in the image. Any kind of movement can interfere with the measurement and the resulting measurement values. Therefore, the series should always be stabilized. Several methods have been used and investigated to achieve the best possible result.

1. Python feature matching using OpenCV

This is an own implementation of image stabilization using OpenCV in Python. A brute force matcher with the SIFT descriptor is implemented for this purpose. The first frame of a series is the reference frame, on which all other frames are aligned. First, all features are identified in the reference image and in the frame which is to be stabilized and then paired with each other using the knnMatch function. The paired features of the two frames can then be used to determine their coordinates. The difference between the X and Y positions can then be used to determine how strongly and in which direction the frame which is to be stabilized deviates from the reference image. This displacement is applied inversely to the frame being aligned so that all vessels and the optical disc are in the same position as in the reference frame.

Once the X and Y shifts have been captured over the entire time interval, three translation methods have been implemented to correct them. One of the methods corrects the absolute displacement values. However, since OpenCV's feature matching can detect the features slightly shifted, as well as in different sizes, sometimes inaccurate or even strongly disturbed displacement values can occur. This can result in partially strong frame twitching which is not caused by any motion. Therefore, two further translation methods have been developed to smooth out the slight to severe translation disturbances: a linear and a polynomial regression through the X- and Y-coordinates respectively (Fig. X).

As soon as the functions for the X and Y coordinates have been determined by one of the three methods, an affine transformation of the frame that needs to be stabilized takes place. No rotation or scale operation is performed, only a linear translation in X and Y direction.   
This method does not require any additional information as input for the image sequence since it identifies the features completely dynamically and pairs them with each other.

1. Hugin Panorama Creator

This free software makes it possible to stabilize a series of images from a camera, for which it is apparently often used in filmmaking communities. After the camera and lens properties such as focal length or horizontal view factor have been configured, image sequences can be imported and further processed. The focus of this software lies on the lens parameters because the imported images are placed on a then generated globe, resulting in stronger affine transformations especially at the edges of the images. It is also theoretically possible to define feature points throughout the sequence, which will be matched to each other in different frames. The output is the stabilized set of images in a separate folder.

1. ImageJ Image Stabilizer by Kang Li

As the name implies, it is a plugin for the free software ImageJ, which is based on Java. It uses the Lucas-Kanade algorithm to stabilize jittery images and can be used for both color and monochromatic images. After the uncomplicated installation of ImageJ, the plugin can also be added without any further adjustments. The image sequence is imported as a stack and the currently displayed frame of the stack is the reference frame to which all others are aligned. The program then estimates the best geometrical transformation of all other slices to match them to the reference frame. Once stabilization has been completed, the stack can be exported from ImageJ as a sequence of frames.

1. ImageJ Template Matching

Another plugin for ImageJ, which comes with a little more setup effort, because it requires additional files in the ImageJ folder. The stabilization algorithm needs a selection of an area to be searched for in all the other frames. For this purpose, a normalized cross-correlation coefficient NCCC between 0 (no match) and 1 (absolute match) is calculated for each pixel in the frames, which need to be stabilized. In this way, the selected area of the reference frame is found in all other frames, so that they can be moved to the position where all selected and identified areas overlap. This NCCC has an inherent correction of different intensities, contrast and lighting conditions. Once stabilization has been applied, the stack can be exported as a sequence of individual images.  
Since mainly this plugin was used for analyzing a large number of sequences (see Results), it has been integrated into an own plugin, which minimizes the effort and the amount of manual intermediate steps. This own plugin takes over the import of an image series as a stack, the preselection of the rectangular selection tool and finally the export of the single images with the corresponding names to a separate folder. This reduces the manual effort to executing the plugin via the menu bar, selecting the first image of the series and selecting the optical disc with the rectangular selection tool. On the one hand, the optical disc is used for image stabilization and on the other hand, the information about the position and size of the disc is stored in a separate file, which is required during the further pipeline.

Warping, rotation, Evaluation tool to compare movements

## Single Image Width Measurement

### Optic disc detection

Resize image

Make b&w mask

Illuminate green channel

Detect disc? = disc center, disc radius

Vessel segmentation

Vessel skeleton, remove branches

Partition eye into zones A, B, C

Vessel width measurement

Spline along the skeleton

Gradient points along the spline, averaging

## Extracting parameters from a Sequence of images

* + 1. Trace the vessel width (FrameTime, remove NaN)
    2. Get vessel parameters
       1. Curve fitting = frequency, bpm, phase
       2. Vessel width = estimate min, max
       3. Vessel elasticity

## Post processing of parameters

* + 1. Outlier identification
       1. By hierarchical clustering (bpm, phase, elasticity)
       2. By thresholds (bpm, elasticity, fitting score)
          1. Reject outliers
    2. Vessel classification
       1. Arteries/Veins by largest phase shift
    3. SQL Database for gathering data
       1. SQL Browsing tool
       2. Verification of vessel parameters (elasticity, plot, vessel class)
    4. Estimate vessel elasticity change across vessels
       1. Vessel width-elasticity plot for each zone
       2. Linear regression through points
       3. Compare plots of zones B and C, esp. of both eyes
       4. Average both zones

1. Classification of elasticity data
   * 1. Big six => prioritize thicker vessel elasticity (threshold for low elasticity)
     2. slope of the function for overall elasticity

# Results

1. Dataset
2. Select from 32 subjects with ca. 2 sequences each (both eyes)
   * 1. Final amount of subjects, vessels etc.
3. Blind dataset (not known, which condition)
4. Task supervision
   * 1. Movement translation
     2. Optic disc detection
     3. Result verification
5. Elasticity
   * 1. …
     2. accuracy
6. Classification
   * 1. …
     2. Comparison with clinical data
     3. Accuracy
7. Runtime
8. Discussion
   1. Does the vessel elasticity measurement support the correlation to diseases?
   2. What is the impact/future of this research project?