**Instructions for AutoMorphalyzer: An updated version of AutoMorph**

AutoMorphalyzer is a fully automatic pipeline which is capable of extracting features of the retinal microvasculature and optic disc from colour fundus photography (CFP). The pipeline utilises fully automatic deep learning methods for segmenting CFP landmarks, including characterisation of the optic disc, cup and classification of the en face retinal vessels into arteries and veins.

AutoMorphalyzer is also capable of extracting clinically-relevant features of interest of the segmented retina and optic disc and cup. The original codebase used to measure features of en face retinal vessels was developed by [Automorph](https://tvst.arvojournals.org/article.aspx?articleid=2783477), whose codebase can be found [here](https://github.com/rmaphoh/AutoMorph). This version has had a major uplift to improve accessibility, usability, and has also fixed some issues regarding feature measurement. At present, AutoMorphalyzer supports regular image files and the open-source codebase can be found [here](https://github.com/jaburke166/AutoMorphalyzer).

**1) GETTING STARTED FROM THE TERMINAL**

The configuration file config.txt is used in conjunction with running the software from the terminal (see below) and contains user-specified parameters. See the table below for each parameter and their definition.

A screenshot of a computer

AI-generated content may be incorrect.

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Description | Expected value | Default value |
| input\_directory | Where AutoMorpahlyzer will look for image files. | Any valid directory path | Path\to\AutoMorphalyzer\test\_images |
| output\_directory | Where the metadata, segmentations and measurements are saved. | Any valid directory path | Path\to\AutoMorphalyzer\output |

**2) RUNNING AutoMorphalyzer ON YOUR DATA**

1. Launch the Anaconda prompt application (see below)
   1. Windows icon PNG and SVG Vector Free DownloadSelect the Windows icon.
   2. Click the drop-down list of the Anaconda3 (64-bit) folder.
   3. Select the Anaconda Prompt (Anaconda3) application.
2. Activate the python environment which stores all python packages necessary to run AutoMorphalyzer.
   1. Type conda activate autommorph-env
   2. Press Enter.
3. Navigate to the AutoMorphalyzer\ directory
   1. Type cd AutoMorphalyzer
   2. Press Enter.
4. Run AutoMorphalyzer.
   1. Type python automorph\main.py
   2. Press Enter.

**3) OUTPUT FILES FROM RUNNING AutoMorphalyzer AT THE TERMINAL**

* In the output\_directory there will be three folders M0, M2 and M3.
  + In M0 the pre-processed CFP images are stored (at dimension (912,912)) with information on how the image was cropped to a square.
  + In M2 the raw segmentations are saved out for the binary-vessel, artery-vein and optic-disc-cup segmentation models.
  + In M3:
    - Composite segmentations overlaid onto the CFP image (saved in M3/segmentations) which can be used for quick inspection of all the output segmentations. Each file (named the same as the original input file), saves out an image of the CFP image stitched together with binary vessels in green, and arteries and veins in yellow and blue, respectively. Additionally, the optic disc and cup are overlaid in green and blue.
    - Feature measurements .csv file saved out.
* In the output\_directory there will be a .xlsx file feature\_measurements.csv storing some simple metadata (filename, centring, laterality, quality) and feature measurements of the disc/cup and en face retinal metrics for each image file row-wise. This file is what should be used for downstream data analysis after batch processing.

**4) CORRECTING SEGMENTATION ERRORS**

* We provide functionality to correct en face retinal vessel and optic disc segmentation via **ITK-Snap**. There are instructions on using ITK-Snap for manual annotations in the directory AutoMorphalyer/manual\_annotations which describe how to use ITK-Snap and correct the binary vessel mask, the artery-vein and the optic-disc-cup segmentation masks.
* Save the manually corrected segmentation masks as .nii.gz files into a new folder called {output\_directory}/annotations. Upon re-running the pipeline again, AutoMorphalyzer should automatically identify any additional manual annotations and re-compute the features!