Architecture Project

A Python program should be implemented until the test meets the following requirements:

|  |  |  |
| --- | --- | --- |
| ID | Description | Comments |
| 5 | After starting the program, two tabs are displayed. The first tab contains video viewing and playback functions (video player), and the second is a video graph of video analytics results. | Tabs are shown |
| 6 | The video display (video monitor) shows a noisy picture with a size of 10x10 pixels after the start | done |
| 10 | A video file in pgm format is interactively selected and opened. Other file formats are not selectable. | Done |
| 15 | Thereafter, other pgm video files can be selected and opened as often as desired | Done |
| 20 | After opening, the tab with the video in the foreground and the first image of the video file always appear in the video monitor of this tab | Done |
| 25 | The size of the video monitor enables the viewing of image details from a distance of 3 m. | TO DO |
| 30 | With a slider any image of the video file can be selected and displayed in the video monitor | Done |
| 40 | A mouse click on a start button plays the video file from the current position until the last picture is reached | Done |
| 45 | The image selection via slider and the playback of the video file can be repeated as often as you like | Done |
| 50 | At the end of the file processing stops | Done |
| 55 | When playing and also when selecting the image with the slider, the horizontal and vertical pupil position of each displayed image is calculated. | TO DO |
| 56 | Another data graph on the video player tab shows the currently calculated pupil position in an ECG-like graph. | TO DO: Requires the inclusion of pyqtgraph |
| 60 | The slider displays the current image position relative to the beginning and end of the file. | Done |
| 70 | The second tab displays three vertically aligned and initially empty data graphs | Done |
| 80 | The data graphs graphically depict the results of the video and data analysis after the playback is stopped | TO DO |
| 90 | The data diagrams show the time on the x-axis and the analyzed quantities on the y-axes. The axis labels indicate the sizes (internationalized) and their units (in the SI system and in square brackets). | Internationalization is extensive; not necessary. |
| 91 | The top data graph shows the horizontal and vertical pupil position in pixel units | TO DO |
| 93 | The middle data graph shows the horizontal and vertical pupil velocity. The "jags" of the quick look jumps (saccades) are removed with a suitably sized sliding (wide) median filter. | TO DO |
| 94 | The lowest data diagram shows the absolute speed incl. View jump spikes | TO DO |
| 100 | The data diagrams can be saved as (vector) image files in common formats (pdf, svg, ...) | Done |
| 110 | The data representation can be enlarged and moved (zoom, pan) | Done |
| 120 | As you zoom in and move the data representation, the timelines of all three graphs remain synchronized | TO DO |

**components**

So far, two major functionalities of the software have emerged:

1. Playback function for video files

2. Display of video editing results

which can also be outsourced to two components or Python modules.

**main component**

The main window remains in the main module and thus forms the switching point between the other modules: the file gui6\_arch\_haupt.py

**Component video**

The video viewing classes are paged out to a separate python gui6\_arch\_video.py file and imported into the main component as Python modules.

**Component analysis**

This component initially displays empty data diagrams. Later, she should see the results.

**Tasks:**

Extend the last Python program with the following functionality:

1. Add another component with its own class where you first store and manage dummy results (such as noise).

2. Implement the requirement 80, but initially only with dummy results. Keep to the specifications for a good architecture.

(a) Tip: use a pandas dataframe as data store for the position and speed data.

3. Implement all other requirements as well

**Image and data analysis**

Build the pupil analysis we did in May into the interactive GUI program. Do not use a loop, but edit one image in the Timer function at a time, save the results in the previously created component and complete the requirement 80 completely. As a reminder, the analysis script is shown here again.