

Software Interface for Visual Field Data Visualization

Rushil Anirudh, July 2009

Abstract: This MATLAB® based interface allows us to view the overall visual field of the patient. The sensitivities (dB) at different points in space are given in an .xls file which serves as the input. It also gives an estimate of how well a patient suffering from poor vision would view a given image.

Procedure:

- I. The sensibility values can be easily plotted¹ at various points corresponding to their spatial coordinates as shown (figure 1). This however, does not allow one to quickly identify the regions with low vision in the given patient.

A different kind of data visualisation was used in this interface, one in which the data is plotted on a map such that the highest value is assigned white [255] and the lowest value is assigned black [0].

A MATLAB® function called `contourf` which plots different levels of intensity was initially used. However, this does not yield satisfactory results as the data gets 'classified' into discrete levels while we require an interpolated set of values (figure 2)

The values of sensitivities in the xls file are given in dB and they need to be converted to normal magnitudes before using them on the Visual Field data visualization. Since the given data has a limited set of values, it is required to interpolate these to achieve good data visualization.

The easiest way around the problem introduced by `contourf` is to use a MATLAB® function called `pcolor` which plots a false coloured intensity map of the given data depending on the resolution defined by the user. To view the results better, a command `shading interp` is used. Similar results can be obtained even without the `shading` command however, they will require a large resolution which leads to more processing and hence more time of execution. The results of `pcolor` are shown below along with the effect of `shading interp`.

¹From Mr. Tien Hophouct's program to display sensibilities

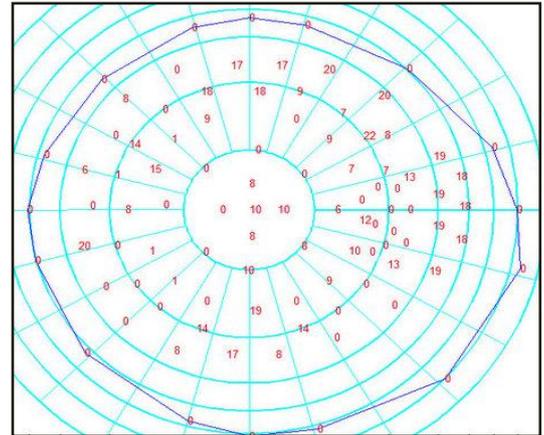


Fig 1: Simply displaying the Values

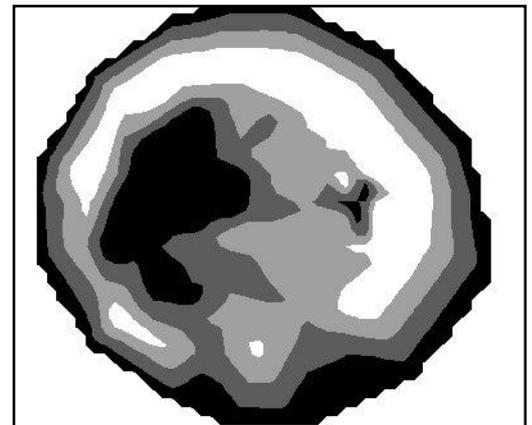


Fig 2: Sensibility with Contourf

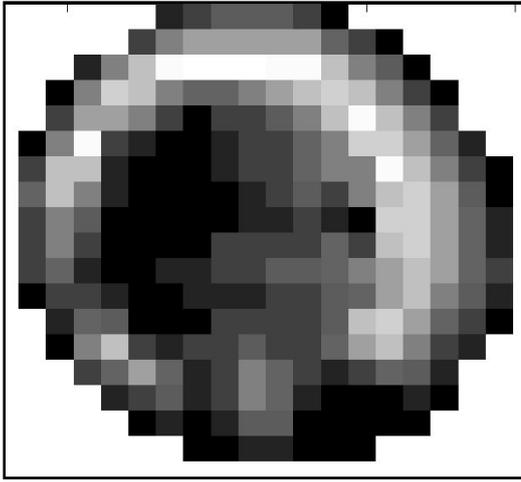


Fig 3: With Low Resolution (20x20 samples)

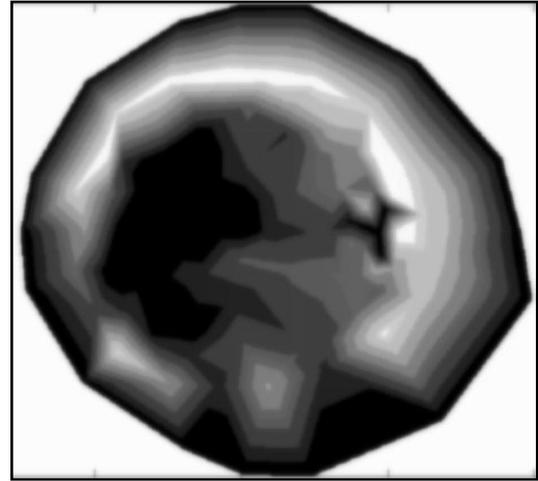


Fig 4: With High Resolution (500x500 samples)

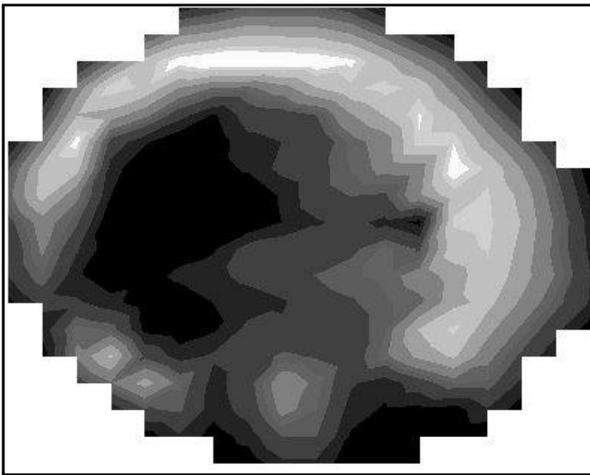


Fig 5: With Low Resolution and Shading Interp (20x20 samples)

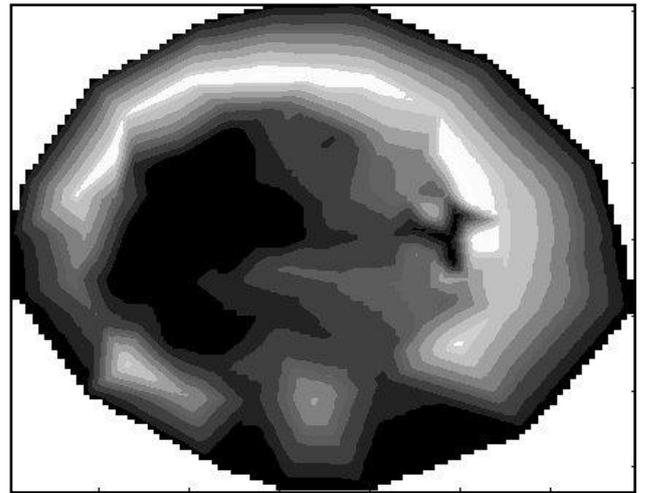


Fig 6: With High Resolution and Shading Interp (100x100 samples)

As seen above, `shading interp` gives us equally good results with lesser processing and time.

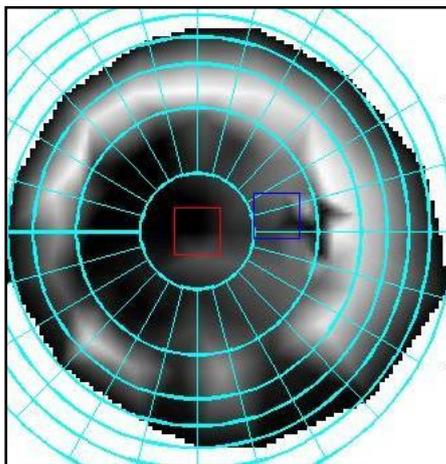


Fig 7: The boxes show the size of the input image relative to the visual field. Red & Blue represent centres chosen by entering values and those by clicking in the required region respectively

II. To display an image with the visual field as a ‘mask’ over it depending on where the image is positioned behind the visual field, the software converts the given image from pixels to degree so as to fit appropriately into the visual field (also in degree) as shown. Since the mask is required to look like its covering the input image, the software converts a small part of the visual field (that which corresponds to the position of the image) into an image. This is followed by normalizing the pixel values of this image such that black = 0; white = 1, and gray takes intermediate values (0-1). To achieve the effect of the mask, every pixel of the input image is multiplied by its corresponding pixel in the visual field image resulting in the “mask over the image”.

The original image is passed through filters, called Spatial Variant Filters, which replicate the retinal compression. This way, the centre of the image is very clear as compared to its edges. This filtering is done after the mask has been applied in accord to the actual retinal model. The results are shown below :



Fig 8: Original Image

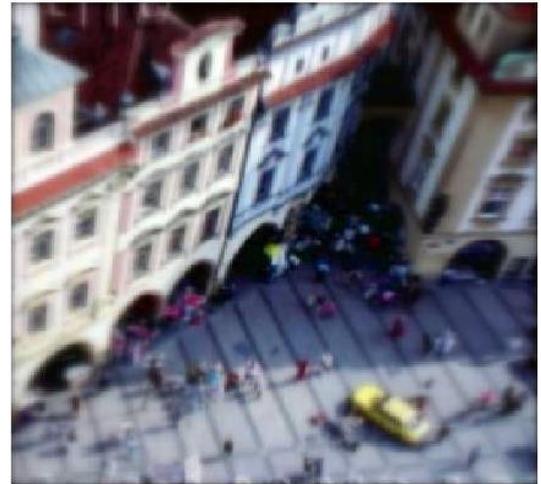


Fig 9: Image after using the Spatially Variant filters

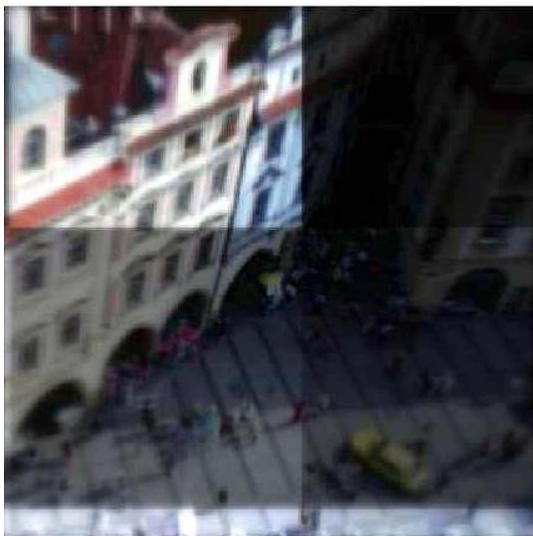


Fig 10: Final Image with 30x30 samples resolution

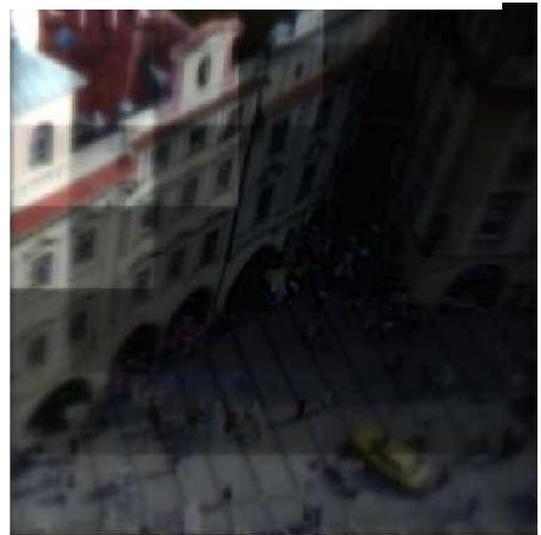


Fig 11: Final Image with 100x100 samples resolution

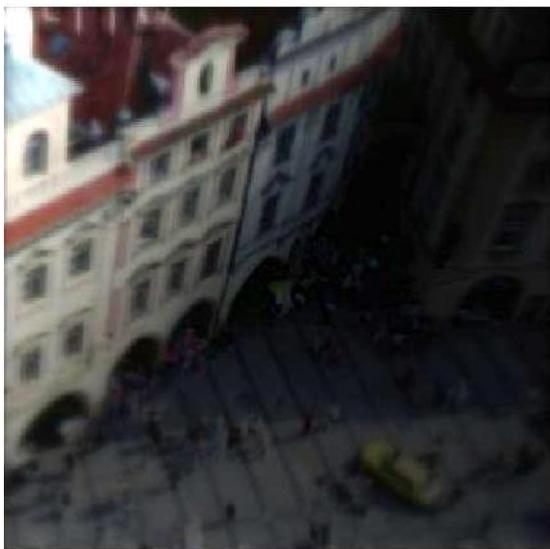


Fig 12: Final Image with 30x30 samples and Shading Interp

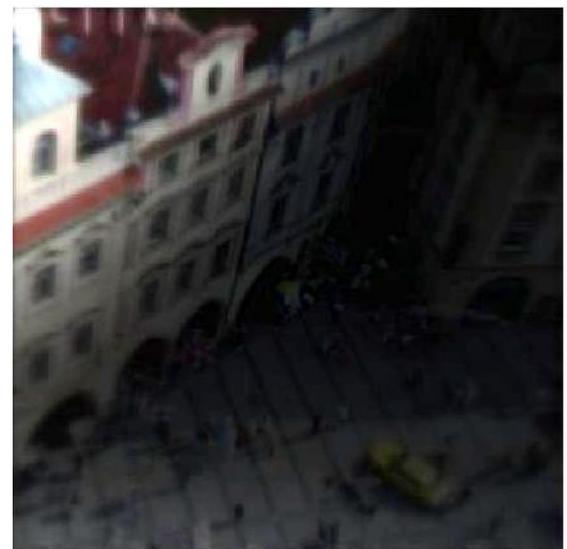


Fig 13: Final Image with 100x100 samples and Shading Interp

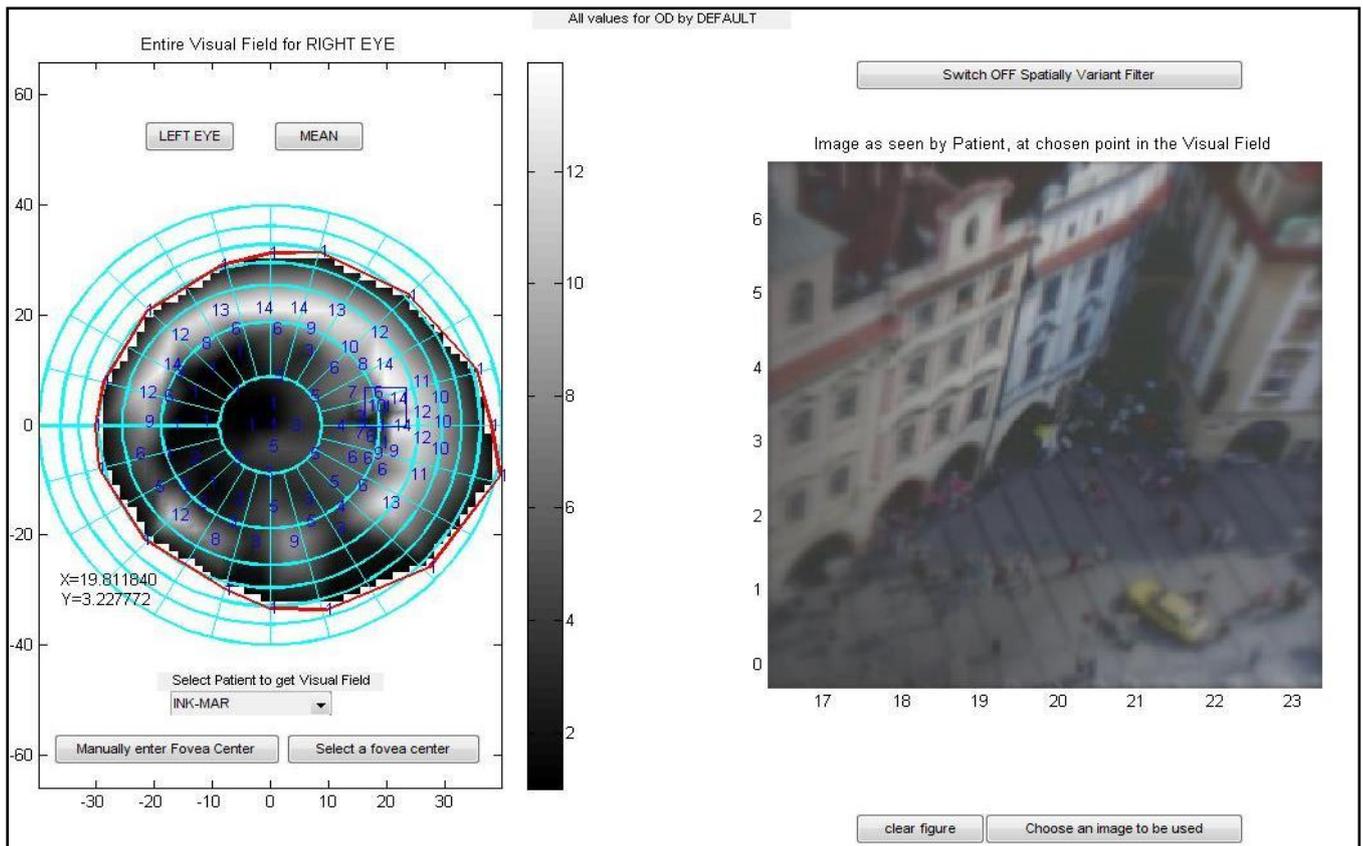


Fig 14: Screen shot of the interface

The above results can be obtained for Left (OG), Right (OD) and a Mean ($\frac{OG+OD}{2}$) of values for all the three patients.