Biomedical Imaging Analysis 4 ICA 1

For this ICA, you should work in a group to create a piece of software to perform an image analysis task of your choice. You should choose one of the datasets listed below to test your software, and write a brief report on your analysis.

The software should be functional and well documented to make it easy for someone with basic programming knowledge (and markers!) to use it.

Q&A

- **Which programming language should we use?**

  Although we ask that you write the central part of the software in Python, you can also use any language of your choice (e.g. R) to perform further analysis, draw graphs for your report etc., if that is more convenient.

  All code should be accessible through GitHub. Although we will only mark the final version, we strongly encourage you to use GitHub as a version control system rather than just using it to save your final version. If you are unsure about how to use it, have a look at this guide to GitHub!

- **Can we use pre-existent software (e.g. FIJI)?**

  In a "real-life situation", there would be no issue in using FIJI or any other pre-written software to perform a task; for this ICA we ask that you develop your own piece of software using Python.

  There are, however, exceptions. For example, you can use pre-existing software to show that your software performs as expected; you can check that the output from the two software match (or that your is better by some meaningful metric!). Or you can use another software to prepare your data in some way. If in doubt, please ask!

- **The task we want to achieve has not been mentioned in the lectures**

  Unfortunately, the course cannot cover all possible topics and techniques related to image analysis. Some of the techniques that you may end up using may be fairly new or not so commonly used to grant having a lecture on them! We encourage you to explore the literature and try new approaches out independently. However, we would like to stress that **there is no requirement for using cutting-edge technology** and you will be able to get full marks even if you do not. Consider very carefully whether the time and effort needed to implement some complex technique will make a difference.

  It is better to have a more straightforward "classical" but working approach than a complex non-functional one.

- **What should our software look like?**

  This is entirely up to you! Examples include:

1. An image-processing software with a graphical user interface (GUI). So, essentially, a window with buttons, sliders, menus etc so that the user can just interact with the software using their mouse, no Python knowledge needed! There are many Python packages to create GUIs. PySimpleGUI is a good one to look at, but there is no restriction on which ones you can use. I will post an example of a simple software with a GUI on Learn.
2. One, or a series of, command-line tools. In this case the user would use it like `python SuperCoolTool.py --input image.png --option1 value1 --option2 value2`.
3. A Python module, containing useful functions to perform a series of related tasks. In this case the user would use it inside a Python script.

Any of these is a valid option, but if you have other ideas, we can discuss them!

- **How should we write documentation?**

Documentation can be as simple as a PDF, or you could generate a simple website.

If you choose to develop a Python module, you could generate some interactive examples, e.g. by creating Jupyter notebooks.

You can include screenshots, videos or whatever you think is necessary. I would encourage you to start developing the content first, then think about the "form".

Note that **we expect the code to be clearly commented as well**.

There is no particular guidance on the length of the documentation.

- **Can I use a different dataset / perform a different task from those provided?**

Absolutely! Be creative, find a publicly available set of images and use them to perform a task of your choice!

I would encourage early discussion of your choice, so that you do not embark on a project that either you cannot complete or that is too simplistic.

I created Slack channels for each group where you can discuss your ideas and ask me questions.

- **What about the summary report?**

The report must be **up to 3 pages long** and briefly describe your software and show examples of the results of running the software on some images. You could use the report to highlight edge cases, to show how the software performs on a particular task or to include some further analysis of the results obtained with your software.

The report should be submitted as a PDF on Learn. Use `BIA4_ICA1_Group_xx.pdf` as a filename.

The submission date/time of the report will be counted as the final submission time of your ICA. Modifications to the code done after this time will be ignored when marking.

No references are needed for this report (although you can add them if you wish).

- **I have some further questions**

Please, use the Slack #ICA channel or the #ICA-group-xx channel to ask any further questions.