

OBJECT ORIENTED PRORGAMMING II – LAB FINAL REPORT

Hikmet Batuhan Görgülü

151220174070

KEY NOTES FOR READER:

- Apart from what is presented in class, Save As Function is added which saves the output image with chosen .jpg or .png extension after browsing the file path. Export as > Source and Export as > Output functions are added which saves the input and output as .pdf, respectively.
- Please run main.py file with cmd after changing directory appropriately. (Running it in VSCode was buggy for me and icons sometimes disappeared)
- There are 3 modules. **Image_Operator.py** is UI generated module, **Backhand.py** is for handling all the backhand connections and image operations and **main.py** is just made for user to intuitively know which module to run for starting the program.
- UI is my original design as required.
- All the menu actions and push buttons have icons, status tips and shortcuts as required
- Project is documented with Doxygen. Go to file Doxygen_Documentation_LABFINAL> html and please open index.html with a browser.

UI:

A screenshot from how UI looks is given below:

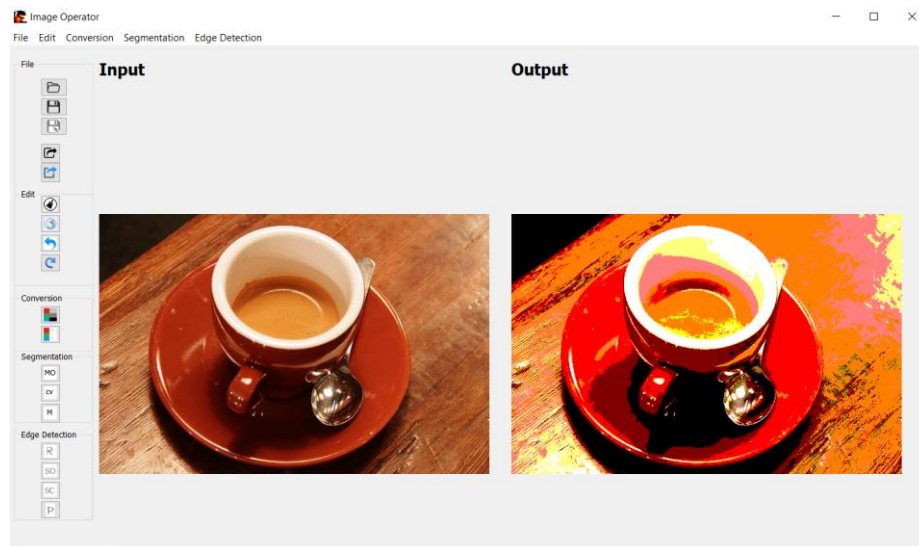


Figure 1: An example UI screenshot

MODULES & CLASSES:

OBJECT ORIENTED PRORGAMMING II – LAB FINAL REPORT

Hikmet Batuhan Görgülü

151220174070

There are 3 modules. **Image_Operator.py** is UI generated module, **Backhand.py** is for handling all the backhand connections and image operations and **main.py** is just made for user to intuitively know which module to run for starting the program.

Since Backhand.py module has class Operate which where all the important methods are. It inherits from QtWidgets.QMainWindow class.

Signal connections, enabling and disabling widgets appropriately and image operating modules exist in this class.

main.py only consists of couple line of codes and its only functions for user to intuitively and easily start the program by just running this file.