README[1] - Implementation For Image Mosaicking

Outcome

- We are able to correctly implement mosaicking feature given the code from our provider.
- We are able to support this feature through script input.
- We are able to expose it through GUI.

Implementation Detail

- By the structure of the project, two classes has to be added (one to `model.image.operation` & the other one to `controller.handler`) for **adding** a new feature, and one existing class (`ImageProcessorMain.java`) has to be modified for **supporting** this feature.

  `MosaicOperation.java` in `model.image.operation`

  - We have to put the underlying mechanisms for mosaicking operation into this class so the model would know what to do on a provided image for this operation.
  - Specifically, we generate a size n list of `SeedNode`s which stores the information of the coordinates on the provided image. We then traverse every `pixel` of the image and find the nearest `SeedNode`s around it. After the traversal, we in effect divide the image into n sub-images (tiles), finally we make each tile have one color of the average of its pixel components and this forms a mosaicking image.

  `MosaicCommandHandler.java` in `controller.handler`

  - We have to add new handler class so the controller would know how to parse the arguments for the mosaic operation.
  - This controller is pretty similar to the BrightenCommandHandler, it receives the first two arguments as the source name of the image and the destination name of the image, with an integer representing the mosaic seed.

  `ImageProcessorMain.java`

  - We need to put an instance of the newly created `MosaicCommandHandler` into Main's `SUPPORTED_HANDLERS` list.

Side Note

Given the prior code is transparent, i.e. the prior implementation detail is visible, the process of implementing a new feature is relatively easy. Especially considering most command controller share the same logic, and the BrightenCommandHandler is a previous instance that expect a series of arguments that is made up of two strings and one integer, as the MosaicCommandHandler does as well.

1. Refer "README (Provider Legacy).md" for the pre-existing design and implementations. ↩