

Assignment 2 Report

Table of Contents

1. Information Briefing
 1. References for Graders
 1. Student names
 2. How to run the program
 3. How to use the system
 2. Code
 1. Starting point
 2. GUI differences from Assignment 2 example interfaces
2. Proof of Performance
 0. Screen dumps of `images/1.jpg` RF initial
 1. Page 1 of retrieval results
 2. Top 20 precision value
 1. Screen dumps of `images/1.jpg` RF iteration 1
 1. Page 1 of retrieval results
 2. Top 20 precision value
 2. Screen dumps of `images/1.jpg` RF iteration 2
 1. Page 1 of retrieval results
 2. Top 20 precision value
 3. Screen dumps of `images/1.jpg` RF iteration 3
 1. Page 1 of retrieval results
 2. Top 20 precision value

1. Information Briefing

1.1. References for Graders

1.1.1. Student names

- Jaimi Chong
- Pranshu Bhardwaj

1.1.2. How to run the program

1. Unzip and open the folder `484 H2 jaimi-pranshu`
2. Ensure the folder `484 H2 jaimi-pranshu` contains the following:
 - `/CBIR.java`
 - `/readImage.java`
 - `/images/*.jpg`
3. Run `CBIR.java` in an IDE of choice for running Java.
 - Program is confirmed to run in VSC and IntelliJ on Windows.

1.1.3. How to use the system

0. While exploring the program, note:
 - Changes to the window title upon applying a sorting method.
 - The enabling and disabling of different `JButtons` and `JCheckboxes`.
 - `ToolTipText` that appears while hovering over disabled sorting method, and image checkboxes.
1. Click the `Next Page` and `Previous Page` buttons to navigate through the image set.
2. Click an image to select it as the target for the next sorting method.
 - It will appear in full resolution under the `SELECTED FOR NEXT JLabel` at the top-right of the window.
 - `Relevance to <selected image>` checkboxes will appear below each image in the grid, but will remain disabled to the user's input until that selected image's first `Sort by: INTENSITY + COLOR CODE` iteration has been run.
 - The sorting method `JButtons` other than Filename sorting will become available.
 - Click another image to change the image selection, or click the same image to deselect it.
3. Click one of the three sorting method `JButtons` at the top-center to display results for the `SELECTED FOR NEXT` image.
 - It will appear in full resolution under the `DISPLAYING RESULTS FOR JLabel` at the top-left of the window.

- Once `Sort by: INTENSITY + COLOR CODE` is selected, note that the previously disabled checkboxes for that image will be enabled, allowing for proper relevance feedback without influence on the first iteration.
 - For that image, the `JButton`'s `JLabel` will update to `(Sort by: INTENSITY + COLOR CODE) && (Update: RF)`. Clicking it additional times will update the weight based on checked images.
 - Note that, since the `JCheckboxes` update based on the selected image regardless of the current `DISPLAYING RESULTS FOR` image or sorting method, relevance feedback can be given for any selectable image at any point in time, so long as it is currently on screen and has had its initial run. This allows for more convenient location of images, such as if a filename was remembered, but not the appearance.
- Once any of the three other sorting methods are applied, the `Sort by: FILENAME JButton` will be enabled. This does not require an image to be selected, and if clicked will remove the `DISPLAYING RESULTS FOR` image.

4. Repeat any of steps 1-3 as desired, or close the program.

1.2. Code

1.2.1. Starting point

We built on our submission for Assignment 1, which was based on the provided `CBIR.java` and `readImage.java` files from `sampleCodeJava.zip`.

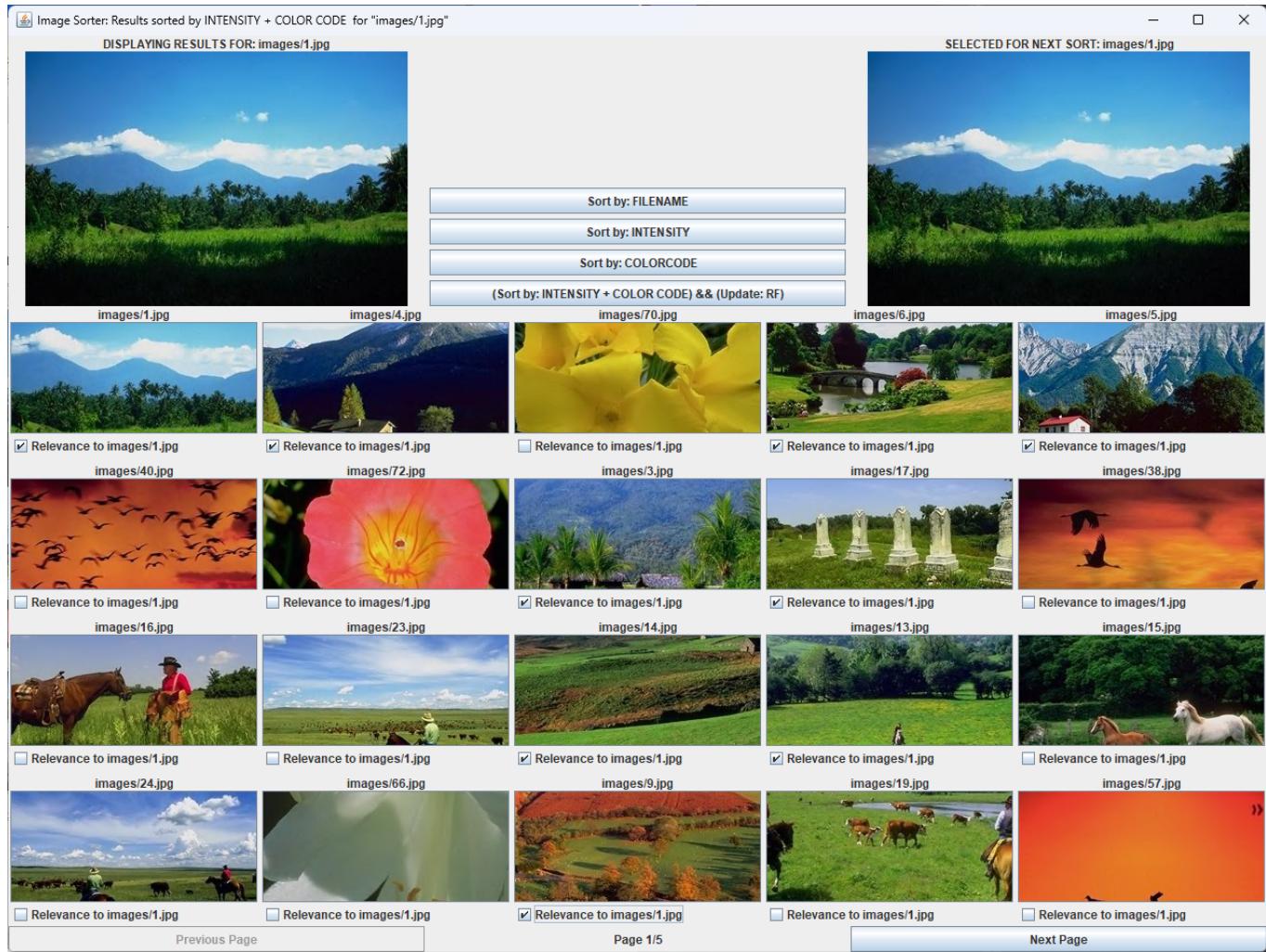
1.2.2. GUI differences from Assignment 2 example interfaces

- Actions
 - Additional sorting methods like `Energy`, `Entropy`, `Contrast`, `Texture`, and combinations other than `Color Code` and `Intensity` are not implemented.
 - Relevance Feedback is not implemented as a `JCheckmark` that enables `JCheckmarks` under images, which presumably auto-update the RF sort upon selection.
- Layout
 - To better distinguish which image the current results are being displayed for, and which image is being selected for the next sorting method application, the latter has been added to the top-right of the window, across from the displayed results image, and both have been given appropriate labels.
 - To better distinguish which image is receiving relevance feedback, additional text specifying which image the checkboxes are relevant to has been added.

2. Proof of Performance

2.0. Screen dumps of images/1.jpg RF initial

2.0.1. Page 1 of retrieval results

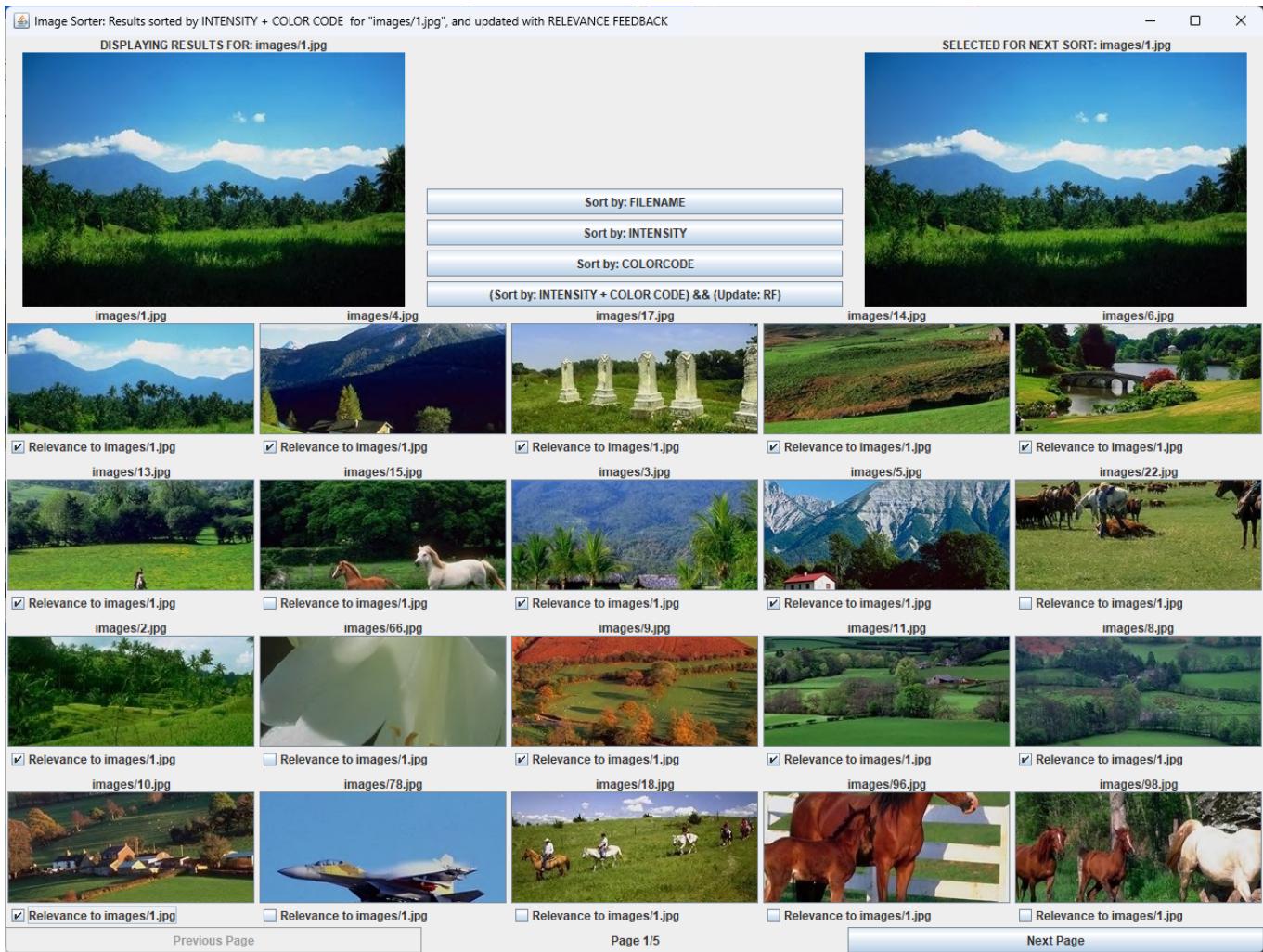


2.0.2. Top 20 precision value

$$9/20 = 0.45$$

2.1. Screen dumps of image/1.jpg RF iteration 1

2.1.1. Page 1 of retrieval results

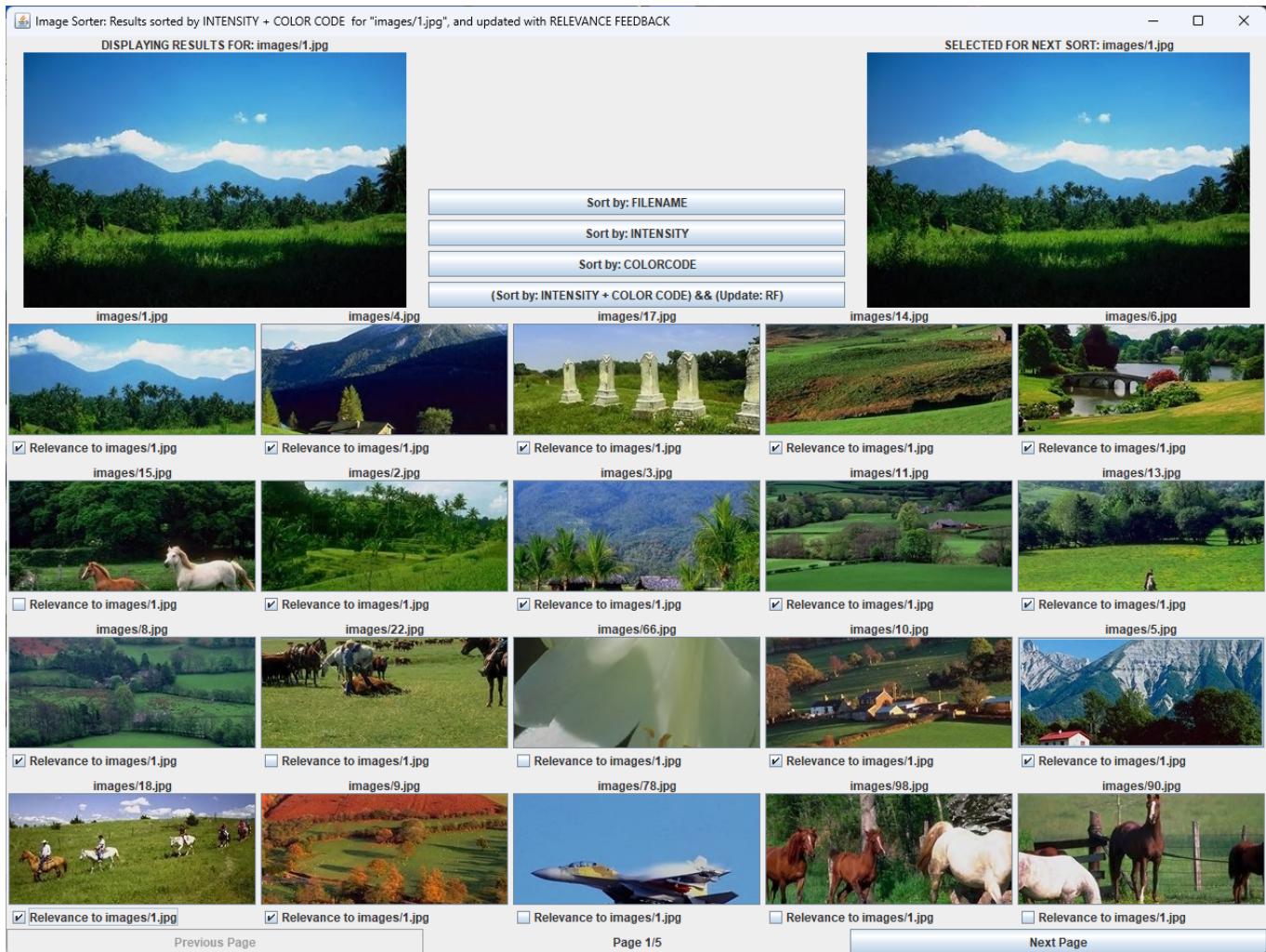


2.1.2. Top 20 precision value

$$13/20 = 0.65$$

2.2. Screen dumps of image/1.jpg RF iteration 2

2.2.1. Page 1 of retrieval results

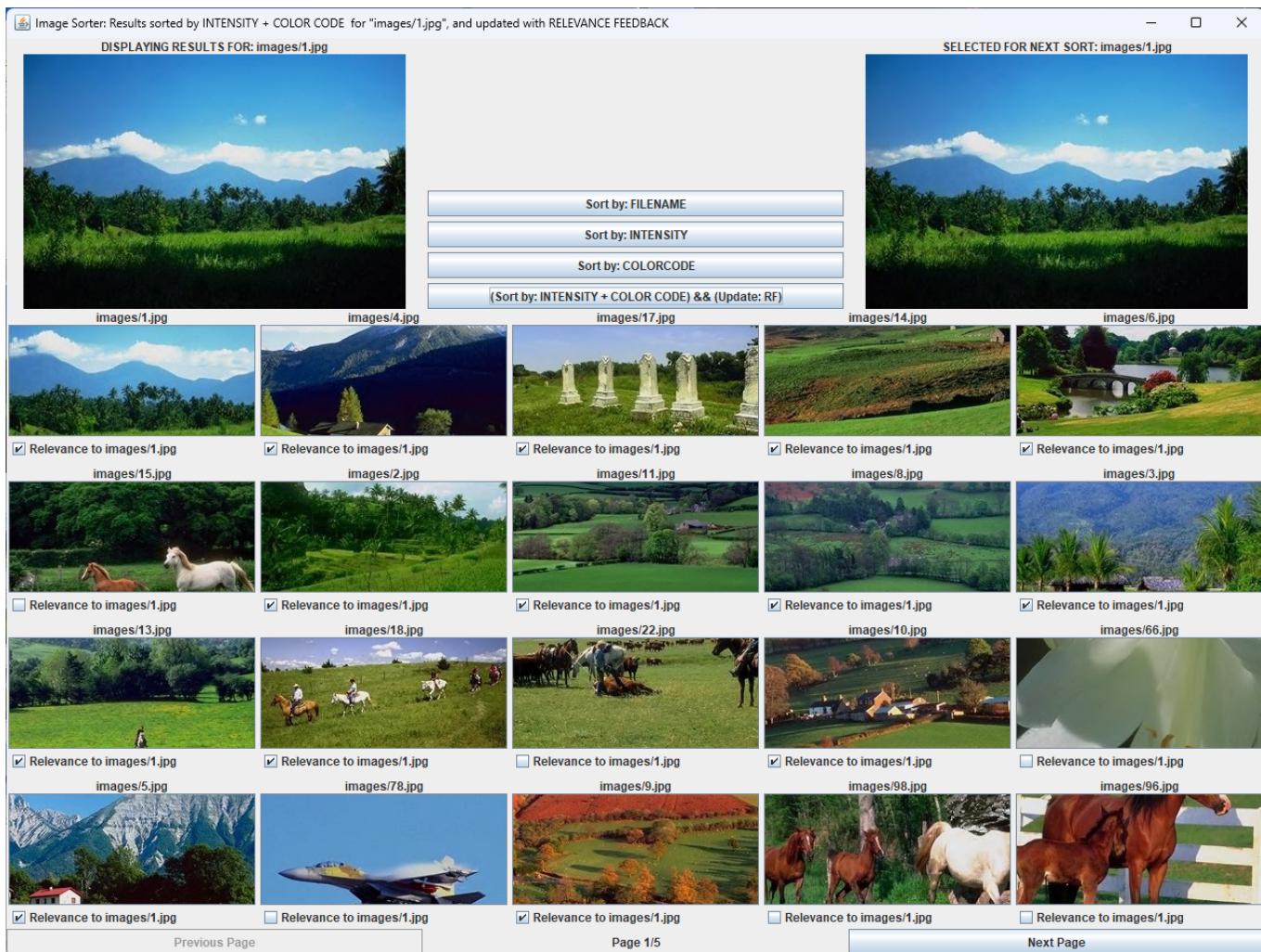


2.2.2. Top 20 precision value

$$14/20 = 0.7$$

2.3. Screen dumps of image/1.jpg RF iteration 3

2.3.1. Page 1 of retrieval results



2.3.2. Top 20 precision value

$$14/20 = 0.7$$