USER STORIES

HDR to Tone Map Converter

10/3/2014

Lynne Lammers

Christine Perinchery

Roxanne Calderon

**Table of Contents**

1. **User Story: Natural Image** ... ... ... ... ... ... ... 3

1.1 Tasks 3

1.2 Tests 3

1. **User Story: Medical Imaging** ... ... ... ... ... ... 4
   1. Tasks 4
   2. Tests 4

**3. User Story: Radar Scatter Data** ... ... ... ... ... ... 5

3.1 Tasks 5

3.2 Tests 6

1. **User Story: Hubble and Other Astronomical Imaging** ... ... ... 7
   1. Tasks 7
   2. Tests 7

**5. User Story: User Interface** ... ... ... ... ... ... ... 8

5.1 Tasks 8

5.2 Tests 8

**User Story 1: Natural Image**

Swapan Chakrabarti wants a way to upload an HDR photo of a natural image and have a tone-mapped image displayed after processing. He wants to enter a file path and have the minimum, maximum, and average intensity of the HDR image displayed. After viewing those values, he wants to be able to enter settings of his choosing for saturation, lightness, and number of tiles. Finally, he wants the tone-mapped image to be displayed, along with the mean, standard deviation, and signal-to-noise ratio of the image. If any of the user-entered values (filepath or parameters) are invalid, an error message will be displayed.

Tasks

1. Take in input for the file path of the image. MATLAB requires single quotes and a .HDR file type. If the file entered is a correct path surrounded by single quotes of .HDR file type, the image will be uploaded to the program.
2. Calculate and print average minimum and average maximum intensity values across the three colors for HDR image.
3. Display recommended and valid parameter ranges. Take in input fields for lightness, saturation, and number of tiles. If input is invalid, MATLAB will issue an error message and the script will exit.
4. Tone map image with user-provided parameters and save image to a variable.
5. Calculate mean, standard deviation, signal-to-noise ratio.
6. Display image alongside mean, standard deviation, and signal-to-noise ratio.

Tests

1. Input: File Path

Tests: 1. Test for correct input, surrounded by single quotes.

2. Test for correct input, without single quotes

3. Test for incorrect a file type that is not .HDR

4. Test for incorrect file path.

Output: Error message or message claiming upload was correct and a new .HDR file

type created.

1. Input: HDR image

Tests: 1. Average, minimum and maximum intensity values calculated

Output: The values are displayed for user.

1. Input: User entered saturation, lightness (two parameters) and number of tiles (2 parameters)

Tests: 1. Test for invalid saturation

2. Test for one and two invalid parameters for lightness

3. Test for one and two invalid parameters for number of tiles

5. Test for correct parameters

Output: Program will accept parameters and display a message or MATLAB will

display an error message and exit .

1. Input: Variables created from user entered parameters

Tests: 1. Test for valid parameters and correct tone mapping.

Output: A file of tone mapped image will be created

1. Input: Tone mapped image.

Tests: 1. Test to ensure mean is a reasonable value.

2. Test to ensure standard deviation is a reasonable value.

3. Test to ensure signal-to-noise ratio is a reasonable value.

Output: Variables created for mean, standard deviation and signal-to-noise ratio.

1. Input: Tone mapped image, mean, standard deviation, and signal-to-noise ratio.

Tests: 1. Test to ensure image and mean, standard deviation and

signal-to-noise ratio appear on screen.

Output: Image is displayed with mean, standard deviation, and signal-to-noise ratio.

**User Story 2: Medical Imaging**

Swapan Chakrabarti needs to view DICOM medical images. To be able to view these on a standard computer screen, however, he needs to tone map the images so that the range of intensities can be displayed. Thus, he would like a program which easily converts these images into a more standard image type. To do so, he will load a .dcm image into the software. From this image, the maximum and minimum intensity will be displayed. After viewing these values, he will be able to enter parameters for the tone-mapping process, including saturation, lightness, and number of tiles. To help guide his selection, a recommended range of values and MATLAB’s valid values will be displayed for all parameters. After he enters the values, the image will be tone-mapped. The mean, standard deviation, and signal-to-noise ratio for the image will be displayed alongside the processed image. If he finds this image acceptable, he can save the image. If not, he will exit the script. If at any point in the process an invalid parameter or file name is entered, an error message will be displayed.

Tasks

1. Take in input for the file path of the image. If the file entered is a valid path surrounded by single quotes and has a .dcm extension, the image will be uploaded to the program.
2. Convert from .dcm to .hdr and save image to variable.
3. Calculate and print minimum and maximum values for image.
4. Display recommended and valid parameter ranges, take in input for lightness, saturation, and number of tiles, and check that user parameters are within the valid range for matlab tone mapping.
5. Tone map image with user-provided parameters and save image to a variable.
6. Calculate mean, standard deviation, and signal to noise ratio for new image.
7. Display image alongside mean, standard deviation, and signal-to-noise ratio.

Tests

1. Input: File path of Image.

Tests: 1. Test for correct input, surrounded by single quotes.

2. Test for correct input, without single quotes.

3. Test for a file type that is not .dcm.

4. Test for incorrect file path.

Output: Program will upload image with message or display a MATLAB error and exit.

1. Input: .dcm file

Tests: 1. Ensure .dcm photo was correctly converted.

Output: HDR file saved.

1. Input: HDR image

Tests: 1. Calculate maximum and minimum intensity of image.

Output: Display to user the maximum and minimum intensity of photograph.

1. Input: User entered saturation, lightness (two parameters) and number of tiles (2 parameters)

Tests: 1. Test for invalid saturation.

2. Test for one and two invalid parameters for lightness

3. Test for one and two invalid parameters for number of tiles

5. Test for correct parameters

Output: Program will accept parameters and display a message or MATLAB will

display an error message and exit .

1. Input: Variables created from user entered parameters

Tests: 1. Test for valid parameters and correct tone mapping.

Output: A tone-mapped image will be created

1. Input: Tone-mapped image

Tests: 1. Test for correct and reasonable mean.

2. Test for correct and reasonable standard deviation

3. Test for correct and reasonable signal-to-noise ratio

Output: Variables containing mean, standard deviation and signal-to-noise ratio

created.

1. Input: Tone mapped image, mean, standard deviation, and signal-to-noise ratio.

Tests: 1. Test to ensure image and mean, standard deviation and

signal-to-noise ratio appear on screen.

Output: Image is displayed with mean, standard deviation, and signal-to-noise ratio.

**User Story 3: Radar Backscatter Data**

Swapan Chakrabarti wants a way to upload a .mat file of radar backscatter data and have a tone-mapped image displayed after processing. He wants to enter a file path and have the minimum and maximum intensity of the .mat file displayed. After viewing those values, he wants to be able to enter settings of his choosing for saturation, lightness, and number of tiles. Finally, he wants the tone-mapped image to be displayed, along with the mean, standard deviation, and signal-to-noise ratio of the image. If any of the user-entered values (filepath or parameters) are invalid, an error message will be displayed.

Tasks

1. Take in input for the file path of the image. If the file entered is a valid file path surrounded by single quotes and has a .mat extension, the image will be uploaded to the program.
2. Convert from .mat to .hdr and save .hdr image to a variable.
3. Calculate and print minimum and maximum intensity values.
4. Display recommended and valid parameter ranges and prompt user to enter values for lightness, saturation, and number of tiles. Store values and check that user parameters are within the valid range for matlab tone mapping.
5. Tone map image with user-provided parameters and save image to a variable.
6. Calculate mean, standard deviation, and signal to noise ratio.
7. Display image with mean, standard deviation, and signal-to-noise ratio.

Tests

1. Input: File path of Image.

Tests: 1. Test for correct input, surrounded by single quotes.

2. Test for correct input, without single quotes.

3. Test for a file type that is not .mat.

4. Test for incorrect file path.

Output: Program will upload image with message or display a MATLAB error.

1. Input: .mat file

Tests: 1. Ensure .mat photo was correctly converted to .hdr.

Output: HDR file saved.

1. Input: HDR image

Tests: 1. Calculate maximum and minimum intensity of image.

Output: Display to user the maximum and minimum intensity of photograph.

1. Input: User entered saturation, lightness (two parameters) and number of tiles (2 parameters)

Tests: 1. Test for invalid saturation.

2. Test for one and two invalid parameters for lightness

3. Test for one and two invalid parameters for number of tiles

4 . Test for correct parameters

Output: Program will accept parameters and display a message or MATLAB will

display an error message.

1. Input: Variables created from user entered parameters

Tests: 1. Test for valid parameters and correct tone mapping.

Output: A tone-mapped image will be created

1. Input: Tone-mapped image

Tests: 1. Test for correct and reasonable mean.

2. Test for correct and reasonable standard deviation

3. Test for correct and reasonable signal-to-noise ratio

Output: Variables containing mean, standard deviation and signal-to-noise ratio

created.

1. Input: Tone mapped image, mean, standard deviation, and signal-to-noise ratio.

Tests: 1. Test to ensure image and mean, standard deviation and

signal-to-noise ratio appear on screen.

Output: Image is displayed with mean, standard deviation, and signal-to-noise ratio.

**User Story 4: Hubble Telescope or Other Astronomical Imaging**

Swapan Chakrabarti has become interested in astronomy. However, many of the images taken by the Hubble Space Telescope and other astronomical imaging devices are of the .FITS file type, which cannot be viewed as an image by most computers. Thus, he would like a program which easily converts these images into a more standard image type. To do so, he will load a .FITS image and have it converted into an HDR file. From this image, the maximum and minimum intensity will be displayed. After he views these values, he will be able to select the parameters for a tone mapped photo including saturation, lightness, and number of tiles. To help guide him along the way, recommended values and MATLAB’s valid values will be displayed for all parameters. After he enters the values, the tone mapped image will be processed. To give him an idea of the quality of this image, a mean, standard deviation, and signal-to-noise ratio will be displayed alongside the processed image. If he finds this image acceptable, he can save the image. If not, he will exit the script and return to the main menu. Likewise, if anywhere in this process an invalid parameter or file name is entered an error message will be displayed and he will be sent back to the main menu.

Tasks:

1. Take in input for the file path of the image. MATLAB requires single quotes and a .FITS file type. If the file entered is a correct path surrounded by single quotes of .FITS file type, the image will be uploaded to the program with a message saying the upload was successful.
2. Extract image from from .FITS file to .HDR image
3. Calculate and print minimum and maximum intensity and display them for user.
4. Display recommended and valid parameter ranges and input fields for lightness, saturation, and number of tiles. If input is invalid, MATLAB will issue an error message and the script will exit.
5. Create tone mapped image with parameters provided by the user and save image to variable.
6. From the newly created tone mapped image, calculate mean, standard deviation and signal to noise ratio.
7. Display image alongside mean, standard deviation, and signal-to-noise ratio.

Tests:

1. Input: File path of Image.

Tests: 1. Test for correct input, surrounded by single quotes.

2. Test for correct input, without single quotes

3. Test for incorrect a file type that is not .FITS

4. Test for incorrect file path.

Output: Program will upload image with message or display a MATLAB error and exit.

1. Input: .FITS file

Tests: 1. Ensure .FITS photo was correctly converted.

Output: HDR variable created.

1. Input: Calculations from HDR file.

Tests: 1. Calculate maximum and minimum intensity of photograph.

Output: Display the maximum and minimum intensity of photograph.

1. Input: User entered saturation, lightness (two parameters) and number of tiles (2 parameters)

Tests: 1. Test for invalid saturation.

2. Test for one and two invalid parameters for lightness

3. Test for one and two invalid parameters for number of tiles

5. Test for correct parameters

Output: Program will accept parameters and display a message or MATLAB will

display an error message and exit .

1. Input: Variables created from user entered parameters

Tests: 1. Test for valid parameters and correct tone mapping.

Output: A file of tone mapped image will be created

1. Input: Tone mapped image.

Tests: 1. Test for correct and reasonable mean.

2. Test for correct and reasonable standard deviation

3. Test for correct and reasonable signal-to-noise ratio

Output: Variables containing mean, standard deviation and signal-to-noise ratio

created.

1. Input: Tone mapped image, mean, standard deviation, and signal-to-noise ratio.

Tests: 1. Test to ensure image and mean, standard deviation and

signal-to-noise ratio appear on screen.

Output: Image is displayed with mean, standard deviation, and signal-to-noise ratio.

**User Story 5: User Interface**

Swapan Chakrabarti wants an intuitive user interface able to upload a variety of HDR images and view the results of tone-mapping. A menu will give him four options for image type: natural scenes, medical images, radar backscatter data, and Hubble telescope or other astronomical imaging. He should be able to select an image type and then proceed through the user story for that image type. He will be able to select a file path for the image. If the file path is incorrect, the script will end. If the file path is correct, the photo will be uploaded. The minimum and maximum intensities will be displayed, and he will be given a recommended range of values for the lightness, saturation, and number of tiles parameters, as well as the valid range. He will be able to input values for each of those parameters. After the image is tone-mapped, it will be displayed with related statistics, and the user will be prompted to save the image to file.

Tasks

1. A menu will display the options for HDR images, along with their respective file types. The user will be able to select which image type they want to use. If the selection is invalid, an error message will be displayed asking the user to enter a valid choice. If the selection is valid, the script for the image type selected will be opened.
2. The prompts for user stories 1-4 to enter a file path will be standardized and clarified to ensure the user enters a correct input. Each user story will still have a specific file type. If an incorrect file type is entered or the path is invalid, an error message will be displayed.
3. The maximum and minimum values of intensity will be displayed for the original HDR image.
4. The prompts for tone-mapping parameters across different file types will be standardized and clarified as needed. The recommended inputs may vary depending on the image type (.hdr, .dcm, .mat, .fits). If an input is given from outside the valid parameters, the user will be asked for that specific input again.
5. The image will be displayed with signal-to-noise ratio, mean, and standard deviation.
6. The user will be given the option to save the file. If the user selects ‘yes’, the user will enter the desired file name, and the image will be saved as a JPEG. If the user selects no, the program will exit. If input is incorrect, the user will be prompted to input again.

Tests:

1. Input: Image type from user.

Tests: 1. User chooses ‘natural images’.

2. User chooses ‘medical images’

3. User chooses ‘radar backscatter data’

4. User chooses ‘Hubble Space Telescope or other astronomical images’

5. User enters incorrect input.

Output: The script associated with user’s selection will be opened or an error message will be displayed and the user will be given the options again.

1. Input: File path

Tests: 1. User enters a file without single quotes.

2. User enters incorrect file type.

3. User enters incorrect file path.

4. User enters correct file.

Output: File type will be opened and saved in the respective user story or error will be

displayed and script will exit.

1. Input: Maximum and minimum intensity calculations from script.

Tests: 1. Maximum and minimum values display in a standardized format.

Output: Maximum and minimum intensity calculations appear on screen.

4. Input: User input for lightness, saturation, and number of tiles

Tests: 1. Test correct lightness.

2. Test incorrect lightness for one and two parameters.

3. Test correct saturation

4. Test incorrect saturation

5. Test correct number of tiles

6. Test incorrect number of tiles for one and two parameters.

Output: Parameters are accepted and saved or user is asked to input incorrect

parameter again.

1. Input: Tonemapped image from script, along with mean, standard deviation, and signal-to-noise ratio variables.

Tests: 1. Image appears in an aesthetically pleasing format along with mean, standard

deviation and signal-to-noise ratio.

Output: Window opens with image and data listed above.

1. Input: User selection.

Tests: 1. User chooses to save photo and gives file path.

2. User chooses to save photo and doesn’t give file path.

3. User chooses to not save photo.

4. User provides invalid input.

Output: The image will be saved as a .jpg file or the image will not be saved and the

script will exit or the user will be prompted for valid input.