



Figure 1: 15-puzzle_scrambled.jpg

ExerciseSheet02

Intro

This assignment is about basic image operations and color images in MATLAB programming.

15-puzzle

In file 15-puzzle_scrambled.jpg you find a ‘scrambled’ version of an image of a 15-puzzle (See Fig. 1). Write a MATLAB script that performs the following tasks (You may want to use `imtool` to find a workflow and then script the operations):

- a) Restore the original version (see Fig. 2) and save it to file 15-puzzle_restored.jpg.
- b) Save a cropped version containing only the puzzle without the background and save it to file 15-puzzle_cropped.png“
- c) Extract all individual tiles and save them to files `tile_1.png`, `tile_2.png`, ..., `tile_15.png` with size 64x64
- d) Create a new image 15-puzzle_reversed.jpg with all tiles of the puzzle in reversed order.
- e) Create a plot of the intensity values on the main diagonal in 15-puzzle_scrambled.jpg.
- f) Create a plot of the minimum, maximum, mean and median (gray-)values of the individual tiles. Include a title, proper annotations of axes, etc.

Color images

Experiment with color channels and models.

- load and display a color image
- separate the RGB channels and separate them separately (in the same figure)
- calculate grey scale image and display (in the same figure)
- convert to indexed image, change the color map and display
- convert to HSV, experiment (and display the results in nicely annotated figures)
 - intensify colors
 - intensify brightness
 - shift colors



Figure 2: Original (greyscale) version of the 15-puzzle image (image from [wikipedia](#))



Figure 3: Original castle image

Vignetting

Create a vignetting effect (See fig.).

- implement a function `vignette` that accpets image and degree of vignetting
- create a figure that shows original and modified images side-by-side
- save modified image as png

Color isolation

Develop a method that allows the user to select a color (or color range) in an image and turn all other colors to grey (similar to Sin City).

- Implement a function `SinCity` for applying the effect
- Let the user select a region of pixel in the original image and retain the mean dominant color in the selected area (+- a possibly selectable band)



Figure 4: Castle image with vignetting effect applied



Figure 5: Original plane image

- create a figure that shows original and effect image side by side
- save the effect image as png



Figure 6: Plane image with Sin City effect applied

Checklist:

- I have used and/or feel comfortable using at least four of the following functions for matrix creation
 - Repmat
 - Linspace
 - Zeros
 - Ones
 - Cat
 - Diag
 - Eye
- I have used and/or feel comfortable using at least four of the following functions concerning matrix dimensions
 - Length
 - Ndims
 - Numel
 - Size
 - Height
 - width
- I have used and/or feel comfortable using at least four of the following functions for matrix reshaping
 - circshift
 - diag
 - flip
 - fliplr
 - fliud
 - reshape
 - sort
 - shiftdim
 - transpose
- I can calculate the following values of an array without using loops:
 - Min
 - Max
 - Mean
 - Median

-
- `Sum`
 - I have frequently used the colon operator (`:`) and `end`
 - I have used
 - `imread`
 - `imwrite`
 - `imtool`
 - `imresize`
 - `imrotate`
 - `plot`
 - `subplot`
 - I have ***NOT*** used
 - For loops (or any other kind of loop)