Introduction to Image and Video Processing
due May 8

Alexia Briassouli (alexia.briassouli@maastrichtuniversity.nl)

Spring 2023

Instructions:

Each of these projects will count for 1/2 of your final grade. They will be checked for plagiarism (software + text). **You are requested to hand in one zip file with title "YourLastName_project1" (e.g. John Doe should submit "Doe_project1.zip").** The zip file should contain:

1. An about 10 page report with your answers to the questions and figures with your results. Page limit is flexible, content matters.
2. The code for producing these results *with clear comments in the code!*. You can use any programming language you are comfortable with (preferably Python or Matlab). You should explain what you think are important parts of the code in the report (e.g. if you use a special trick that you are proud of).
3. You should include references and starter code links that you used (see lecture 1 slides for details).

Your grade will depend on how clearly you present and explain your results in the report and code. You are allowed some freedom to explore solutions (e.g. if you are asked to come up with your own method), so *there is no one correct answer*. However, you should demonstrate you have understood the class material and how it applies to these projects.

1 Color Spaces

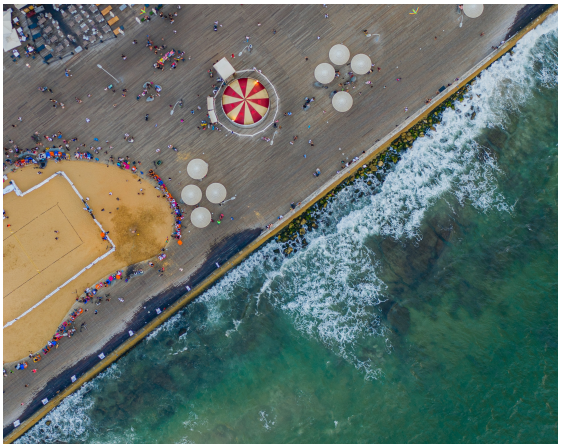
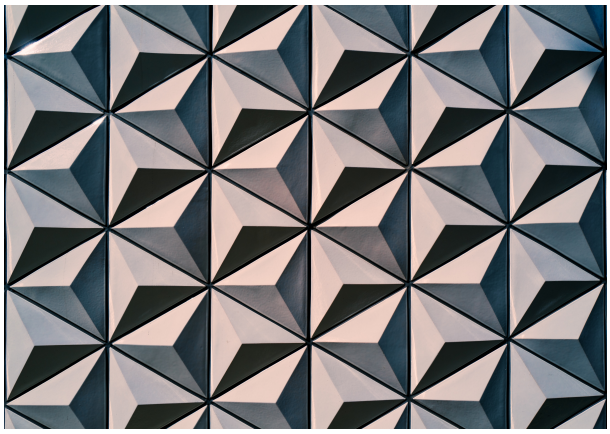
Choose two color images, one with high and on with low contrast. You can use the ones below or yours.


1. Transform your 2 images from RGB to HSV and display your results. Display each color channel (R, G, B and H, S, V, i.e. 6 in total) and explain the appearance of each channel with respect to your image properties (e.g. S has high values where the saturation is high etc).

2. Apply histogram equalization to the H, S, V channels separately. Plot the resulting histogram, the histogram equalized channels, and the resulting color image. Discuss the results.
3. Apply histogram equalization to the V channel only. Plot the resulting histogram, the histogram equalized channels, and the resulting color image. Discuss the results.

2 Edge Detection

Use the image below or an image of your liking with strong 45°, 135° diagonal edges.

1. Create a 45°, 135° 1st order spatial edge detection filter. Apply it to the image using the filter2d inbuilt function and binarize the filtering result. Display and discuss the result.
2. Add salt and pepper noise to the image. Apply the previous edge detection filter. Display and discuss the result. Apply a de-noising filter and then apply the previous edge detection filter. Display and discuss the result.
3. Add Gaussian noise to the image. Apply the previous edge detection filter. Display and discuss the result. Apply a de-noising filter and then apply the previous edge detection filter. Display and discuss the result.

3 Fourier transform

Use the image below, or any one you like with **no** strong periodic features.

1. Create a 2D cosine or sine of same size as the image, with a frequency of your choice **only** in the horizontal or vertical direction using meshgrid. It can have an amplitude of your choice. Display and discuss its appearance.

2. Find its 2D FFT and display the FFT magnitude centered in 2D. Display a 1D slice of the FFT magnitude from the middle row and column (2 plots) of the FFT magnitude. What does it show? Discuss all figures.

3. Create an image corrupted by periodic noise by adding the 2D cosine/sine to your image. Calculate the FFT of the resulting noisy image and display its magnitude centered in 2D. Display a 1D slice of the FFT magnitude from the middle row and column (2 plots) of the FFT magnitude. What does it show? Display and discuss all figures.

4. Create a frequency domain filter to remove the periodic noise from the previous question's noisy image. Display and discuss: (1) the de-noising filter's FT magnitude in 1D and 2D, (2) the de-noised image's FT magnitude in 1D and 2D, (3) the resulting de-noised image.

5. Repeat the last 2 questions (3.3, 3.4) with periodic noise of the same frequency but a different magnitude. Display and discuss: How does this affect the de-noising result? Why?

4 Special Effects

1. Use the image below or an image of your own with regions that are textured but have mainly one color. Perform color segmentation on the image using K-means. Transform the segmented image into a cartoon by adding black outlines using methods learned in class.

2. Use the image below or an image of your own, preferably with a grid-like pattern Create a geometric transform in polar coordinates that transforms the coordinates as follows:

$$\phi' = \phi, \ \rho' = \sqrt{\rho}$$

Use inverse mapping to create the transformed images. Display and discuss your results.