**IPV Assignment 2 (week 2)**

**Week 2\_1**

**Goal:** Create Live Scripts for future demo or assignment submission

## Create Live Scripts

In Matlab, choose ‘help’ and search key words: **Create Live Scripts**, then follow the steps to create your live scripts document for your week1 assignment.

## Submit your week2 assignment via canvas

1. Make a zip fold and name it as:

e.g., Week2\_ Team1\_John (1 student);

Week2\_Team2\_Coenraad & Milen(2 students);

1. Your zip fold should include **your .mlx file** and **image(s)** your used for this assignment.

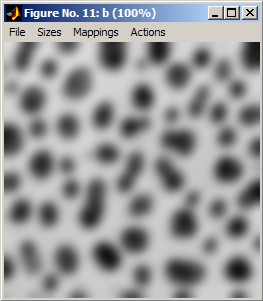
**Week 2\_2**

# **Goal: Basic Image filtering**

## Blurring filters(smoothing)

☞ Finish the part 5 from <ftp://qiftp.tudelft.nl/DIPimage/docs/Introduction_to_DIPimage.pdf>

Choose “Gaussian filter”. The name between parentheses on the menu indicates the name of the function that implements this filter. The required input image should already reside in one of the variables, e.g. **a**. Type any name for the output image, for example **b**. Now we need to choose the size of the Gaussian filter: the standard deviation in pixels. Try out different values for it, and see what happens.



**Q1:** Also explore the “Uniform filter”. What is the difference with the “Gaussian filter”? What parameters would you choose to make the result of the uniform filter similar to the result from the Gaussian filter? Why can’t you make the results exactly the same?

## Filter Images Using imfilter

☞This example shows how to filter an image with a 5-by-5 filter containing equal weights (often called an averaging filter) using imfilter.

MATLAB® has several two-dimensional and multidimensional filtering functions. The function filter2 performs two-dimensional correlation, conv2 performs two-dimensional convolution, and convn performs multidimensional convolution. Each of these filtering functions always converts the input to double, and the output is always double. These other filtering functions always assume the input is zero padded, and they do not support other padding options.

In contrast, the imfilter function does not convert input images to double. The imfilter function also offers a flexible set of boundary padding options.

Read image.

I = imread('coins.png');

**Create filter.**

**h = ones(5,5) / 25;**

Apply filter to image using imfilter.

I2 = imfilter(I,h);

Display original image and filtered image for comparison.

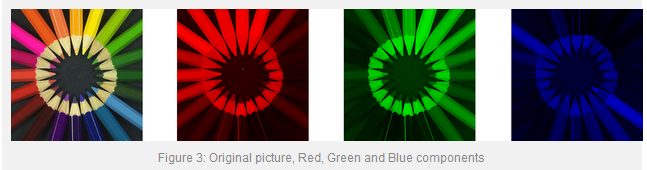
imshow(I), title('Original Image');

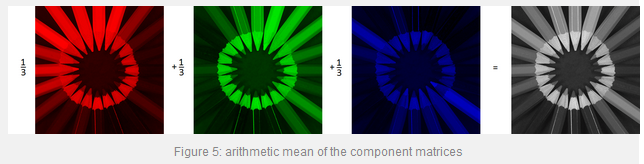
figure, imshow(I2), title('Filtered Image')

**Q2:** Try out the above example and understand it, then design your owner **kernel or mask** and apply it to your image. And please add explanation for your filter design and final filtered image.

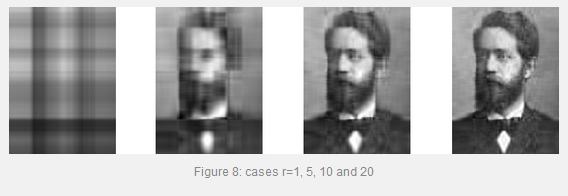
**Q3:** Read this paper <http://dmuw.zum.de/images/6/6d/Matrix-en.pdf> and understand it. Then implement **at least 2 types of operations** in matlab(you can choose your own images):

**Operation: explore RGB**





**Operation: Filter**



**Operation: matrix transformations**

