Project 3:

Option 1:

(a):

Challenges:

1. Number of layers can be provided optionally. But, if the given number of layer is not suitable with the image size, then it will be overwritten by calculation. It is calculated by log formula.
2. Image size must be of size of 2^k. For this reason after selecting the image size, the images must be padded to make that size. In the below code I have used cop-edge padding.
3. For anti aliasing, a gaussian filter with sigma = 2 has been used.

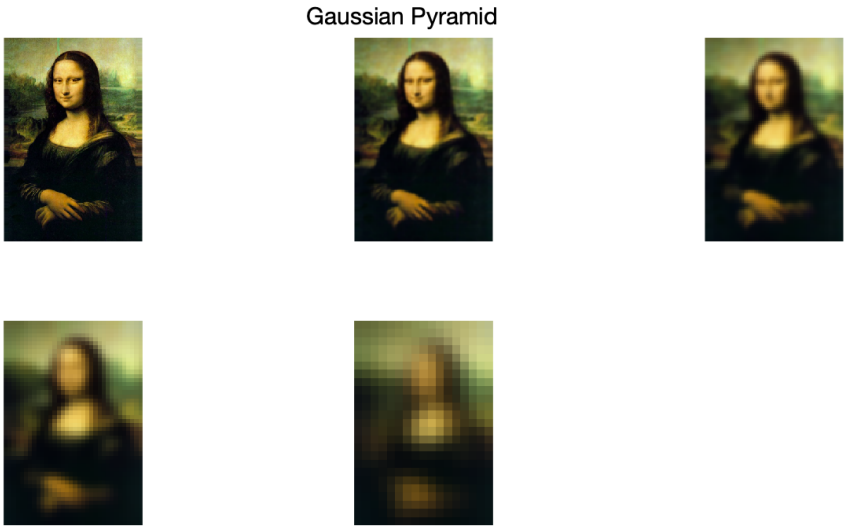
Gaussian and Laplacian pyramid:

```
function [gauss_pyr, laplacian_pyr] = compute_pyr(input_image, num_layer)

    % input_image RGB or grayscale image (H, W) or (H, W, 3)
    % num_layer layers to be calculated in the pyramid
    % total num_layer will be num_layer + 1 including original image
    % if num_layer doesn't satisgy with size of input_image, num_layer..
    % will be overwritten by best avialable option

    % gauss_pyr cell array of length num_layer+1 (Gaussian pyramid)
    % laplacian_pyr cell array of length num_layer+1 (Laplacian pyramid)

    H = size(input_image,1);
    W = size(input_image,2);

    min_legth = 16;
    max_layer = min(floor(log2([H W]) - log2(min_legth)));
    num_layer = min(num_layer, max_layer); % overwrite num_layer if needed
    num_layer = num_layer + 1; % including the original image

    % calculate optimal image size and add padding
    lowest_size = [H W] / 2^(num_layer - 1);
    lowest_size = ceil(lowest_size);
    padded_size = lowest_size * 2^(num_layer - 1);
    padded_img = my_pad(input_image, padded_size(1), padded_size(2), "copy-
edge");

    % for anti-aliasing
    gauss_ker = fspecial('gaussian', [10 10], 2);
    %surf(gauss_ker)

    gauss_pyr = cell(1,num_layer);
    gauss_pyr{1} = padded_img;
    for k = 2:num_layer
        img = gauss_pyr{k-1};
```

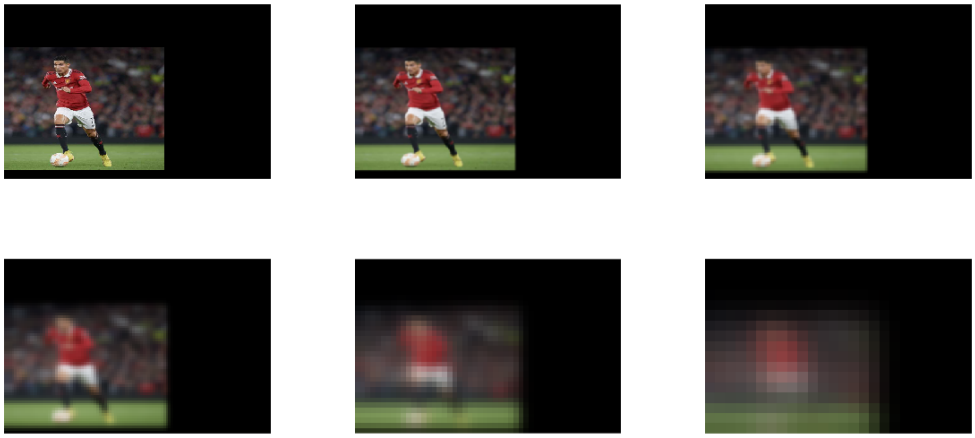
```
        gauss_ker_freq = fspecial('gaussian', [size(img,1) size(img,2)],
100);
        [~, img] = my_conv2(img, gauss_ker, "copy-edge", "same"); % smoothed
        gauss_pyr{k} = my_downsample(img, 2);
    end

    laplacian_pyr = cell(1,num_layer);
    laplacian_pyr{num_layer} = gauss_pyr{num_layer};
    for k = 1:(num_layer - 1)
       high_res = gauss_pyr{k};
       low_res = my_upsample(gauss_pyr{k+1}, 2);
       laplacian_pyr{k} = high_res - low_res;
    end

end
```

Result:

Example of Gaussian pyramid:

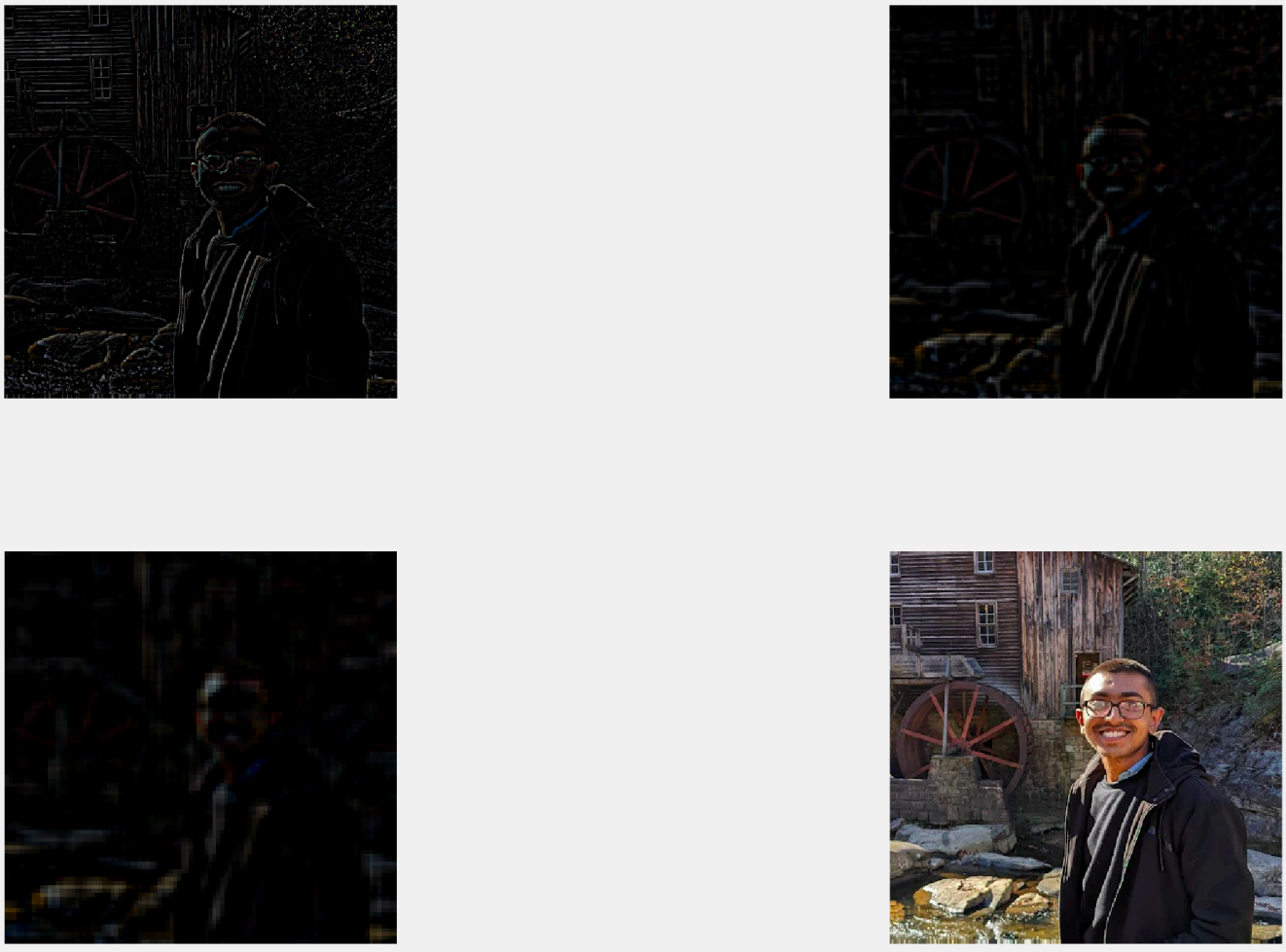


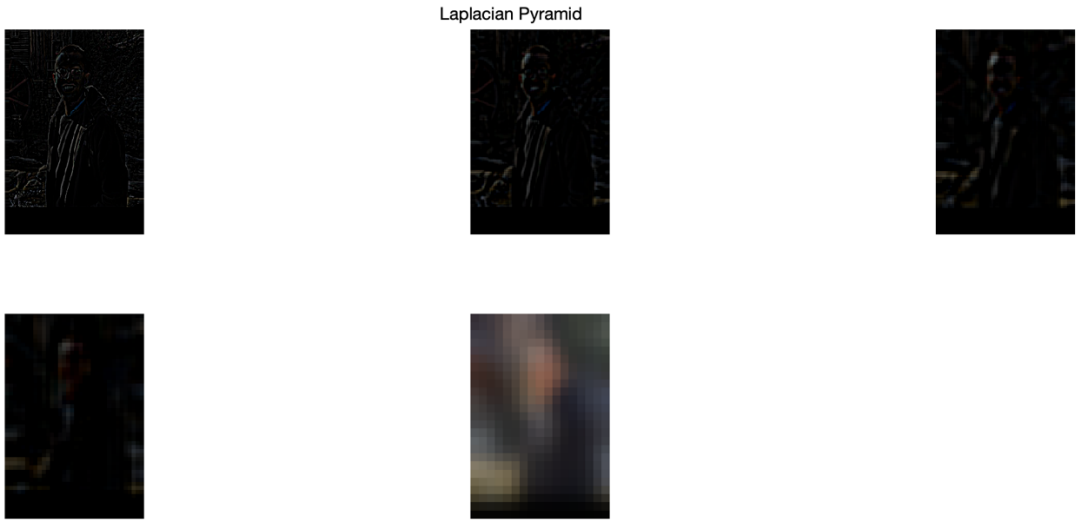
Gaussian Pyramid



Example of Laplacian pyramid:

Laplacian Pyramid





(b)

GUI:

I have found two options for GUI in matlab:

The first one has many option, and we can just draw a contour to draw mask. In the 2nd option we need to brush over all of the area that we want to select.

But, the 1st one has a disadvantage that it requires a lot of memory, sometimes for very high resolution images it takes a very long time. In that cases, I have used the second option.

1st option:

```
addpath("make_mask_fcn_v2_file_exchange/") % option 1

im_con = imread('ex1/monalisa.png');
D = make_mask_fcn_v2(double(im_con)/255);
uiwait(D.d.fig)
con_mask = mask;
close(D.d.fig)
```

2nd option: This code saves the mask as png files as png file.

```
addpath("GUIImageMaskSample/") % option 2


optipon 2 (alternative) use this if option 1 seems to be slow
GUIImageMaskSample() % and then browse image -> draw mask and hit save
im_con = imread('ex1/monalisa.png');
con_mask = imread('ex1/monalisa_msk.png')>0;
im_tar = imread('ex1/my_photo.png');
tar_mask = imread('ex1/my_photo_msk.png')>0;
im_con = imread('mona.png');
```

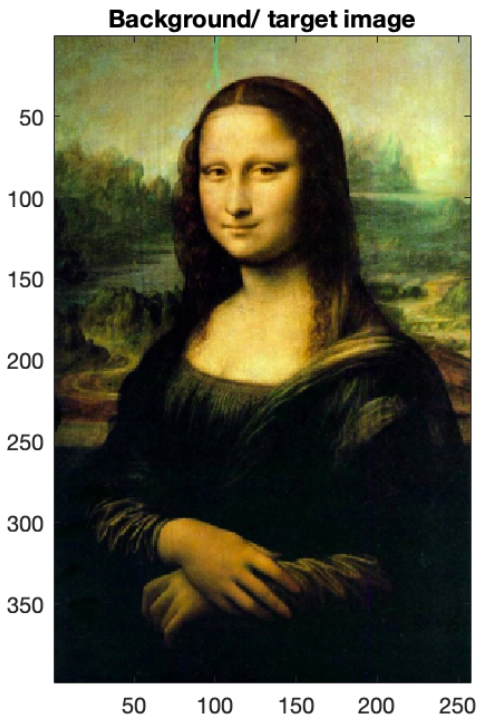
(c) Image blending:

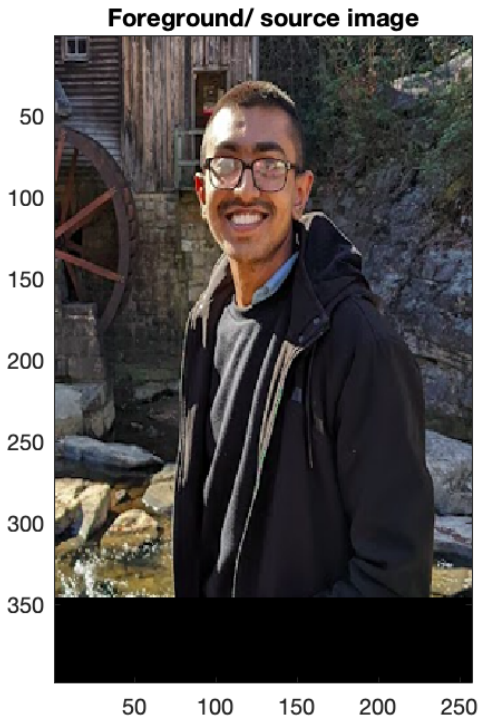
Challenges:

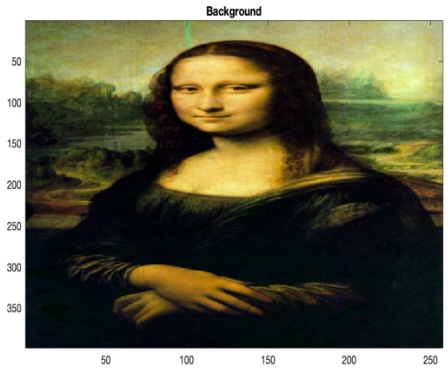
1. Alignment: 1st the sizes are matched to of the two images using the bounding boxes of the two. Then the position is aligned using desired the overlapping region's co-ordinate.

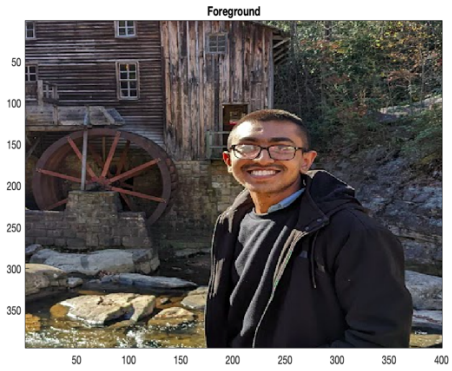
Results:

Example 1: Monalisa and me (face alternate)

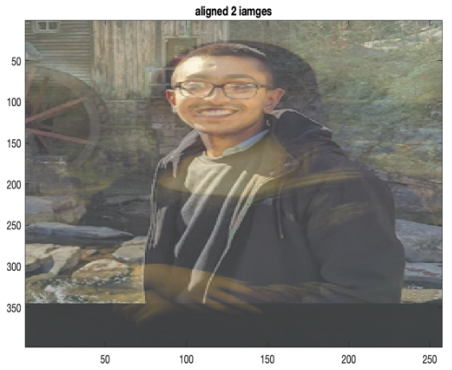


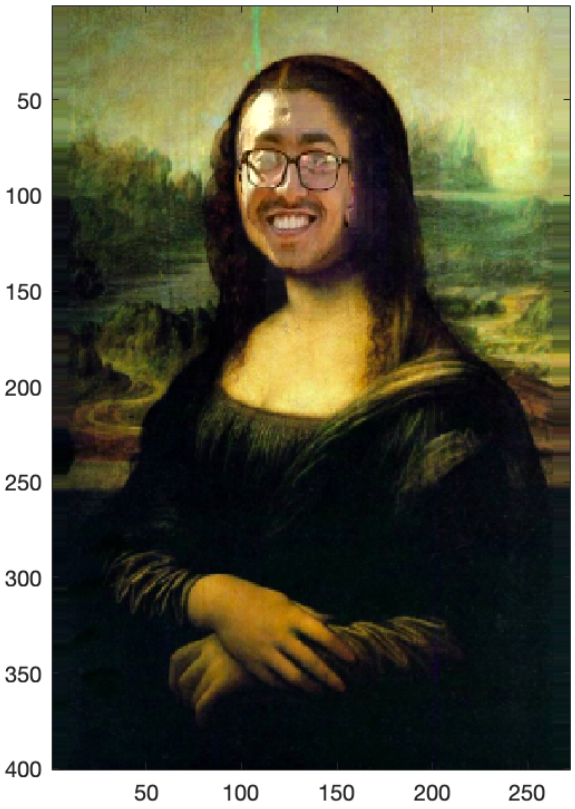



Background
50
100
150
200
250
300
350
50
100
150
200
250


Foreground
50
100
150
200
250
300
350
50
100
150
200
250
300
350
400


scaled foreground
50
100
150
200
250
300
350
400
50
100
150
200
250
300

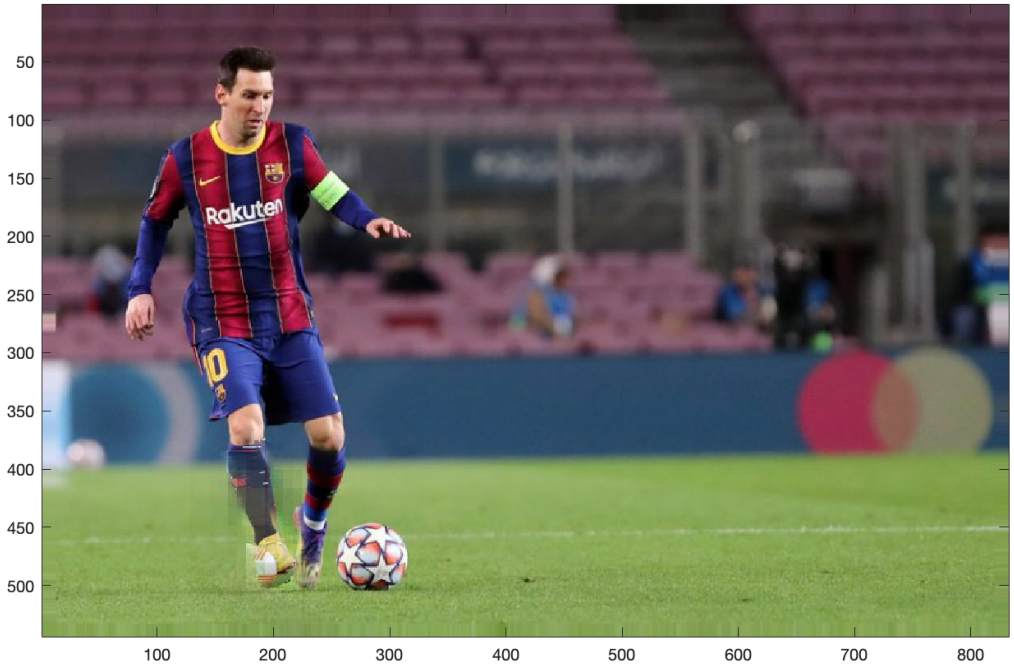

aligned 2 iamges
50
100
150
200
250
300
350
50
100
150
200
250



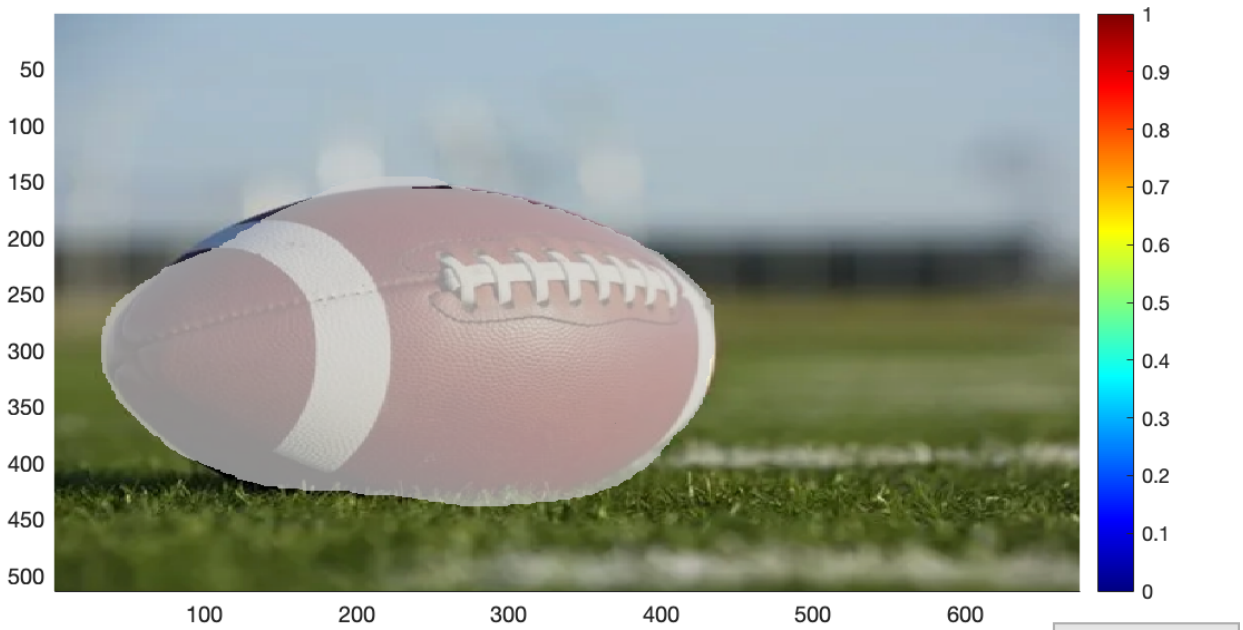
Exapmple 2: Leg swap of messi and Ronaldo

Zoomed version: Alignment of 2 legs (perfect!)

Blending result:



Example 3: Swap cricket ball with Rugbi



Options [right click]

<--Undo

DRAW

Redo-->

-> <- Erode

<- -> Dilate

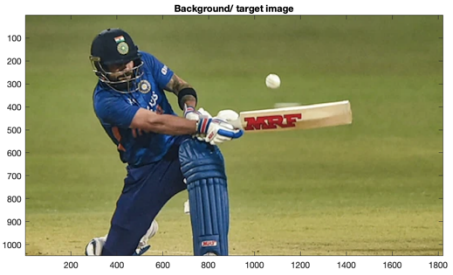
Invert

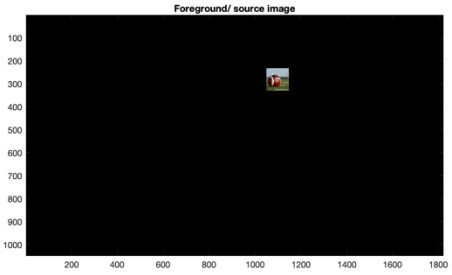
Save To Workspace as "mask"

L-click = +mask, R-click = -mask
wheel = grow/shrink-mask
shift+wheel = rotate mask

Clear Mask

Size: [1,1], angle: 0





Blended result