Technion - Israel Institute of Technology

Hw_2 - Hough Transform and Laplacian pyramid

Chen Katz, ID 203511043

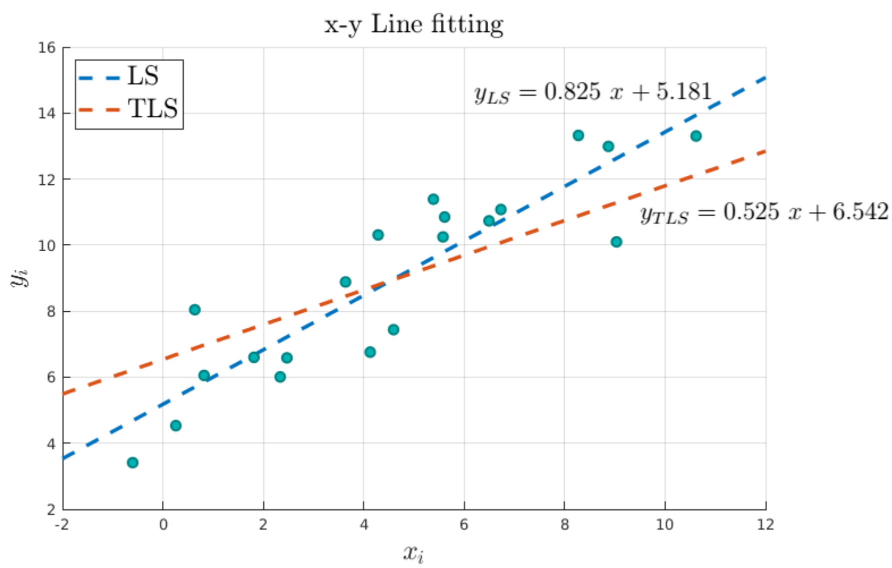
Daniel Engelsman, ID 300546173

---

1 Question

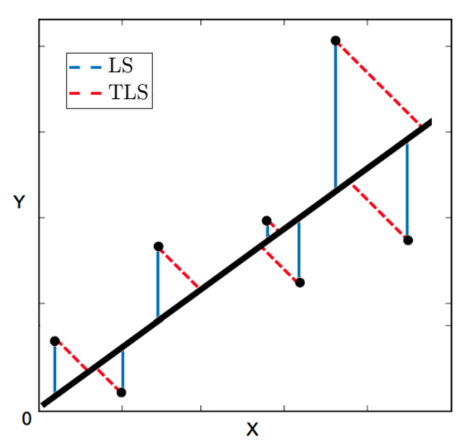
i. Using Least Squares (use the Matlab function `pinv`). What are the values of the slope and the intercept

ii. Using Total Least Squares. Use the formulation described in the appendix in the tutorial section (Q-and-A.pdf file). What are the values of the slope and the intercept?



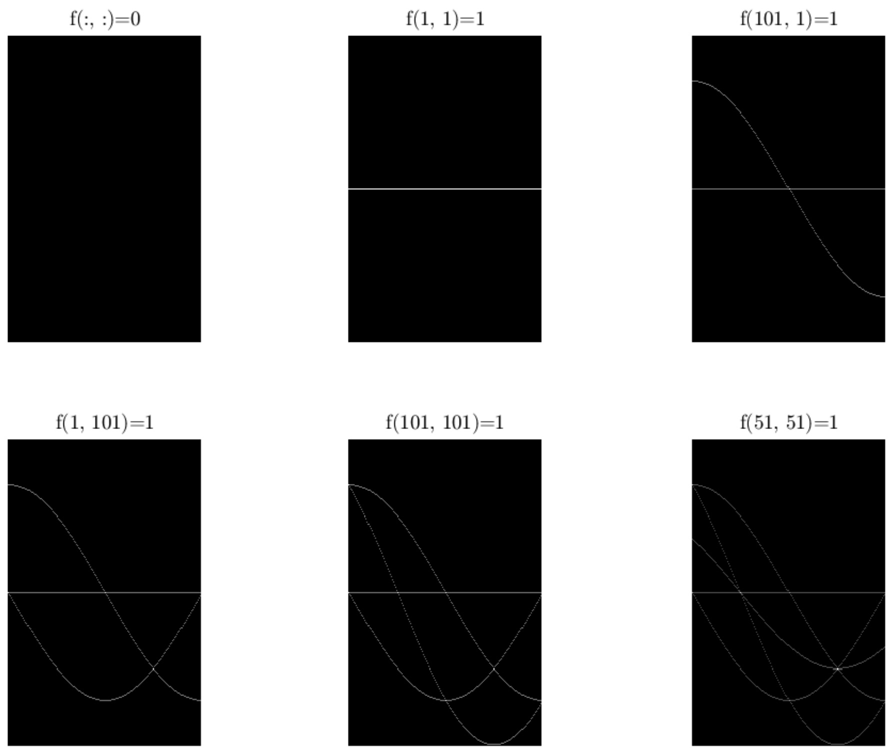
Explain the results : **LS** assumes that only one variable is free of error while **TLS** assumes both variables are subject to error (= more realistically). Therefore, each cost minimizes differently the sum of distances, expressing different slope and intercept (above) :

$$\mathrm{J}_{LS}: \ \arg\min_{\beta} \|\mathbf{X}\beta - \mathbf{y}\|^2 \quad \text{vs.} \quad \mathrm{J}_{TLS}: \ \arg\min_{\beta} \Big\| \big[\, (\mathbf{X} + \Delta x)\beta \ , \ -(\mathbf{y} + \Delta y) \,\big] \Big\|^2$$



2 Question

a. Explain the result of each *imshow* as it appears on the Hough domain :



The Hough transform (=HT) implements the following (r, θ) parametrization :

$$y = -\frac{cos(\theta)}{sin(\theta)}\, x + \frac{r}{sin(\theta)} \quad \Leftrightarrow \quad r = x\ cos(\theta) + y\ sin(\theta)$$

$(\ \cdot\ )$ $f(:,\, :) = 0$: The f-matrix is completely blank and no values are to be transformed.

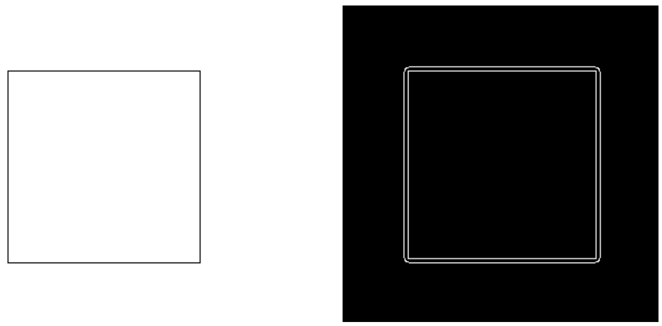
$(\ \cdot\ )$ $f(1, 1) = 1$: a point on the image space's <u>origin</u> $(0, 0) \rightarrow$ horizontal line of $r = 0$.

$(\ \cdot\ )$ $f(i,\, j) = 1$: any further point on the image plane is translated to the Hough space using the above (r, θ) transformation, and added to the subplots.

Note that in *Matlab* the image origin is located at the top-left corner.

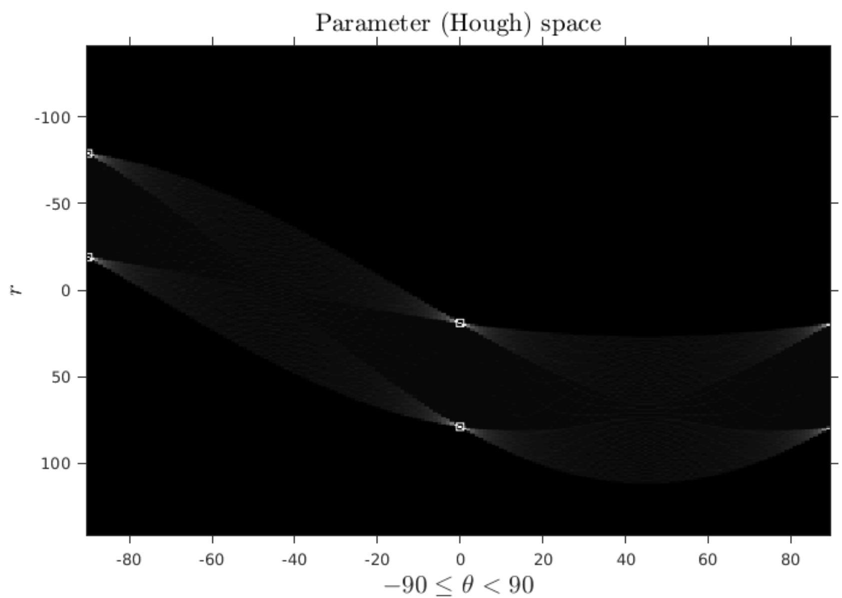
b. Generate a binary image of a square &

c. Use the *canny edge detection* to generate an image containing the edges of the square :



Left : binary square Right : after edge detector

d. Display the Hough transform of the image, and explain the results :

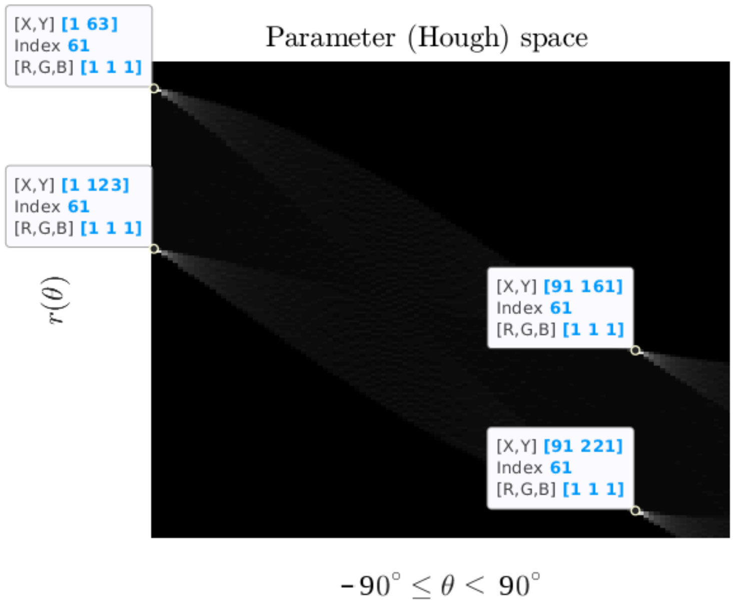


Four perpendicular lines correspond to four HT graphs using (r, θ) parametrization such that cells with the highest intensity denote the line equations that fit in the x-y plane.

e. Why lines in image space are represented as maximum point of the Hough transform ?

The Hough space can be seen as a vote matrix of $(r,\ \theta)$ combinations, for each $(x_i,\ y_i)$ point in the image plane. After plotting the respective curves in the Hough space, we'll seek for intersections ($\Leftrightarrow$ local maxima) that represent the most voted model (x-y line).

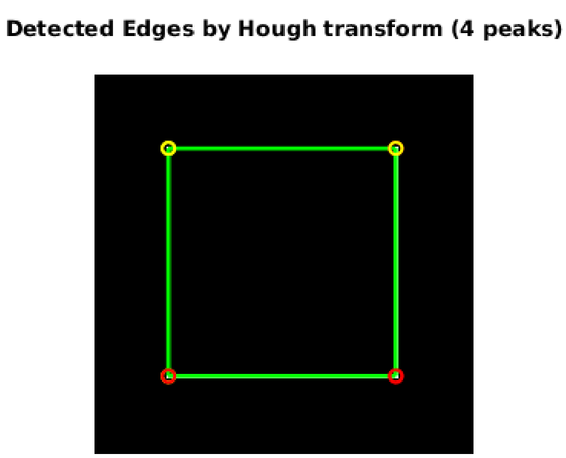
f. Use the function *houghpeaks* on the Hough transform of the square :



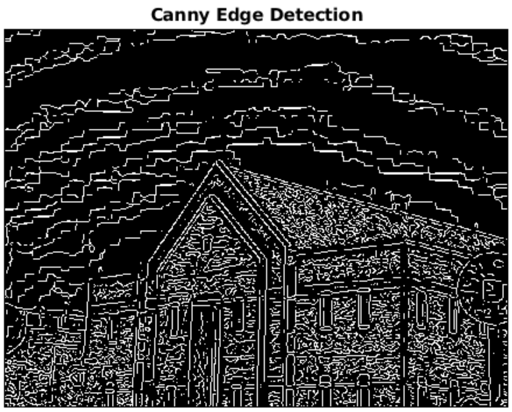
The function returns the cells with the highest votes for the x-y line equations :

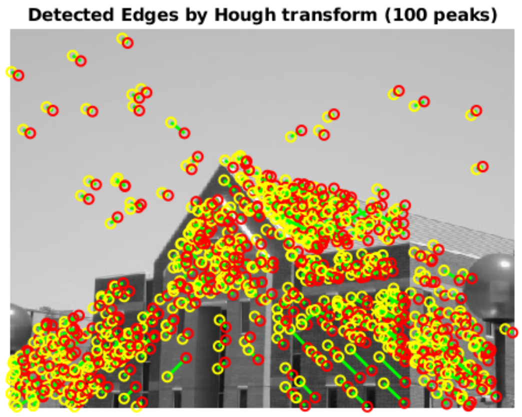
$$
\begin{aligned}
R_{up} &= x\ \cancelto{0}{cos(-90^\circ)} + y\ sin(-90^\circ) &\Rightarrow\quad y_{up} &= R_{up}\\
R_{down} &= x\ \cancelto{0}{cos(-90^\circ)} + y\ sin(-90^\circ) &\Rightarrow\quad y_{down} &= R_{down}\\
R_{left} &= x\ cos(0^\circ) + y\ \cancelto{0}{sin(0^\circ)} &\Rightarrow\quad x_{left} &= R_{left}\\
R_{right} &= x\ cos(0^\circ) + y\ \cancelto{0}{sin(0^\circ)} &\Rightarrow\quad x_{right} &= R_{right}
\end{aligned}
$$

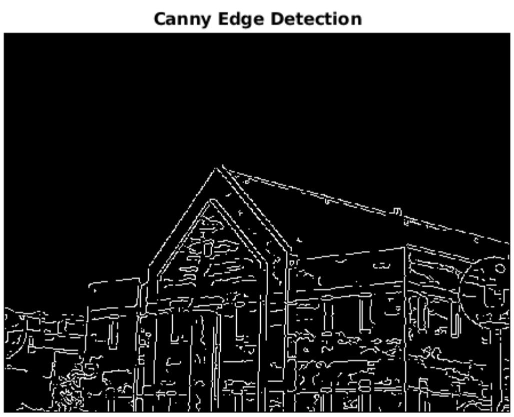
g. Use the function houghlines and plot the detected lines over the square image :

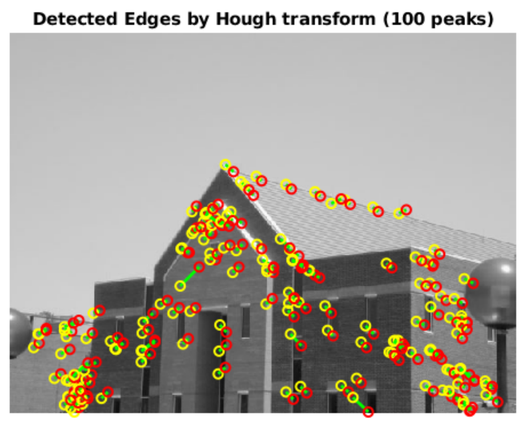


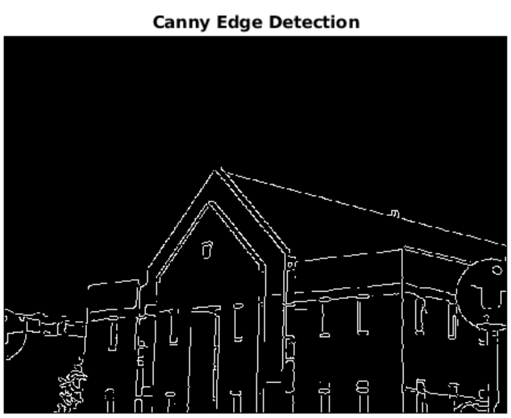
h. Reproduce the results for input image *building.jpg* , using different canny edge values : The following images show several CE threshold values of - [0.01, 0.05, 0.15].

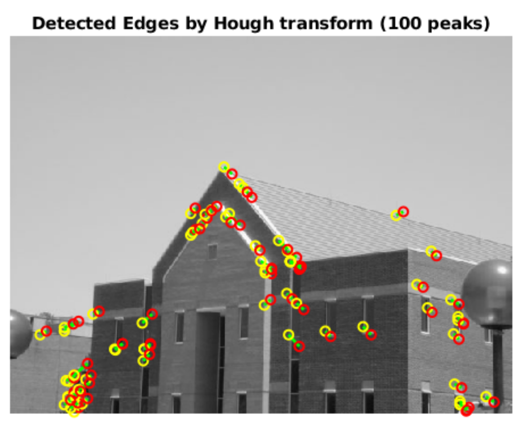












We can see that lower CE threshold present more detected edges since more peaks are detected. Contrarily, higher threshold reduce the peaks and we get less detected edges.

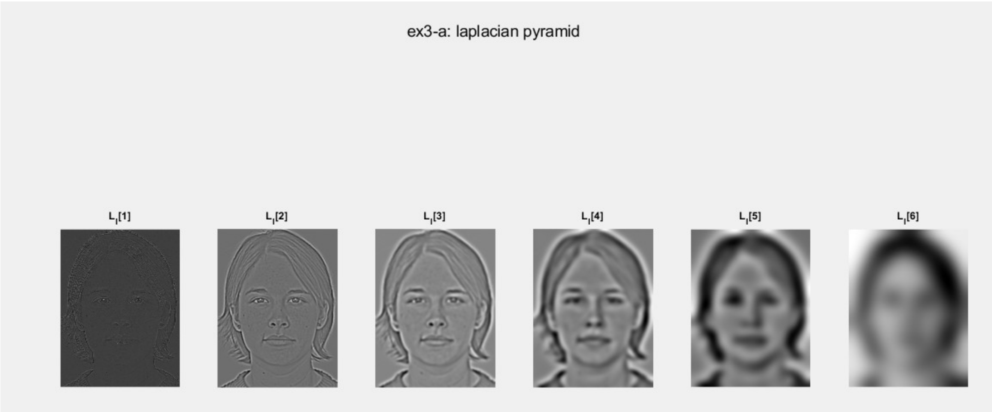
Question 3:

3-a)

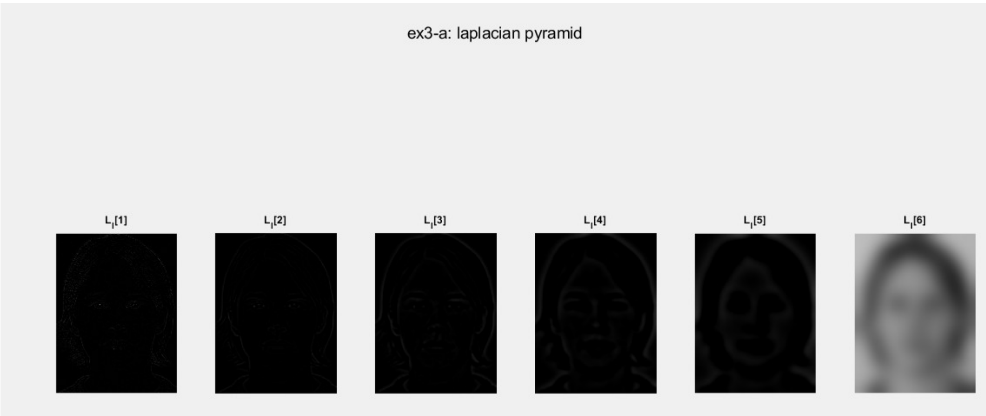
Example - the output of the function that decomposes a gray-level image to its Laplacian pyramid:

```
-number of levels = 6
-input = grayscale of '6.png' image
```

Version 1 – display the images with full display range (`[min(I(:)) max(I(:))]`):



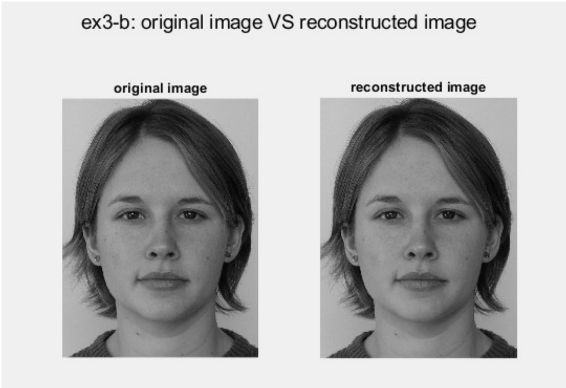
Version 2 – display the original images:



Explanation:
we extracted the image features such as edges at multiple scales
$L_l[1]$ – contain High freq. $L_l[5]$ – contain low freq. $L_l[6]$ – contain the residual

3-b)

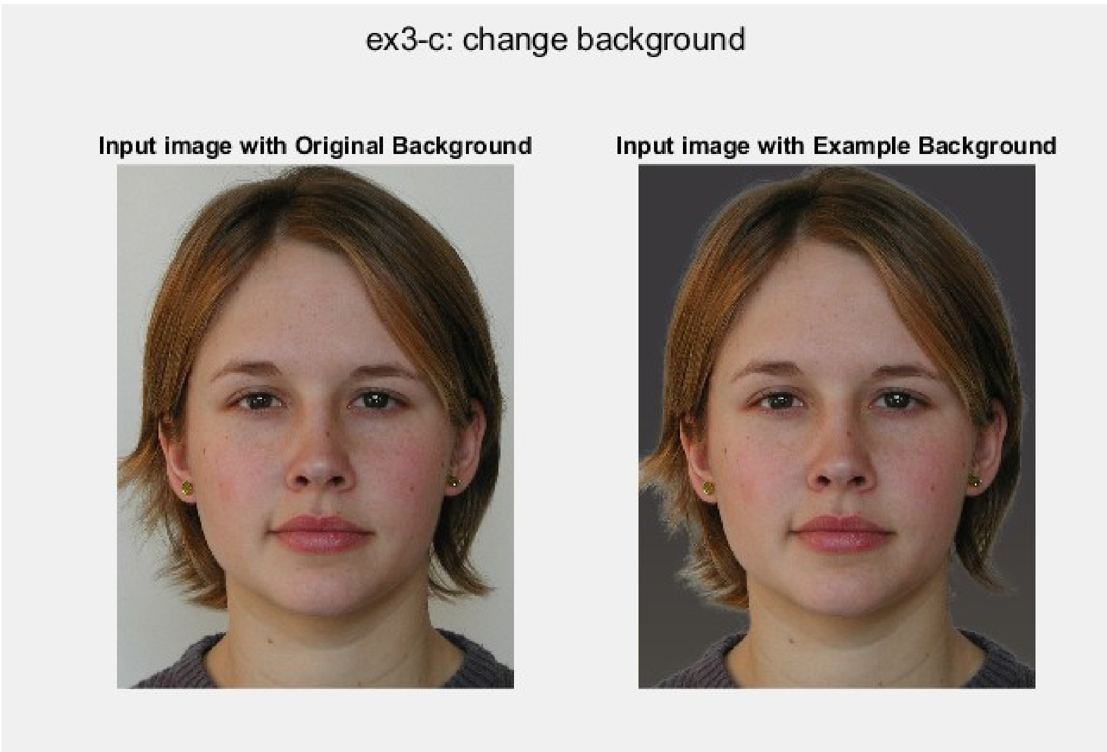
reconstruct its Laplacian pyramid using the function of section (a) with n=6 levels:



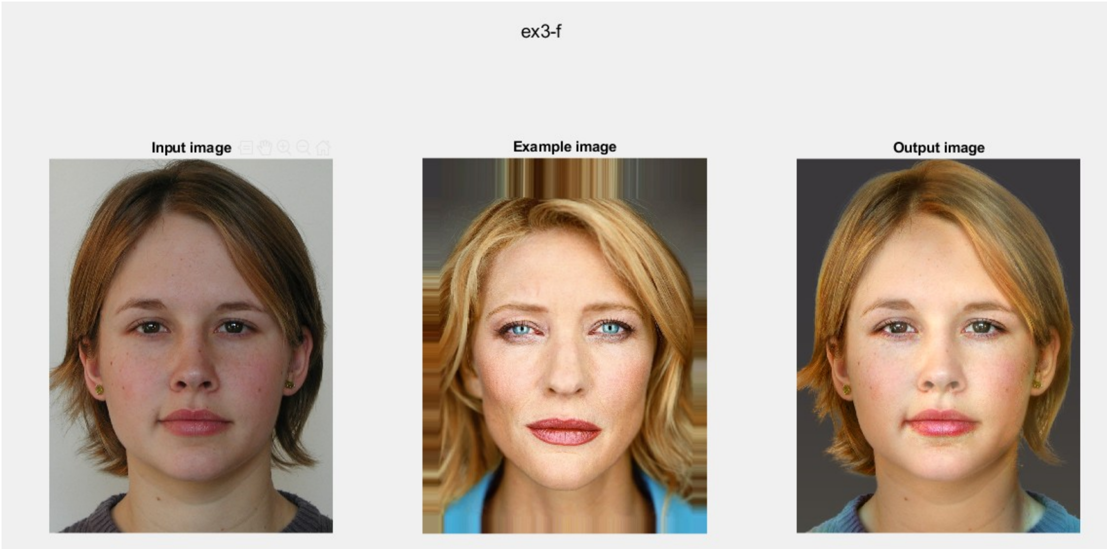
-reconstruction with down-sampling in not accurate and that is because that down-sampling causing a lost of information doe to the "expand" process (in the expend process we evaluate the 'lost' pixels values and that is not accurate)

reconstruction without down-sampling in accurate, you can see it from formulas.

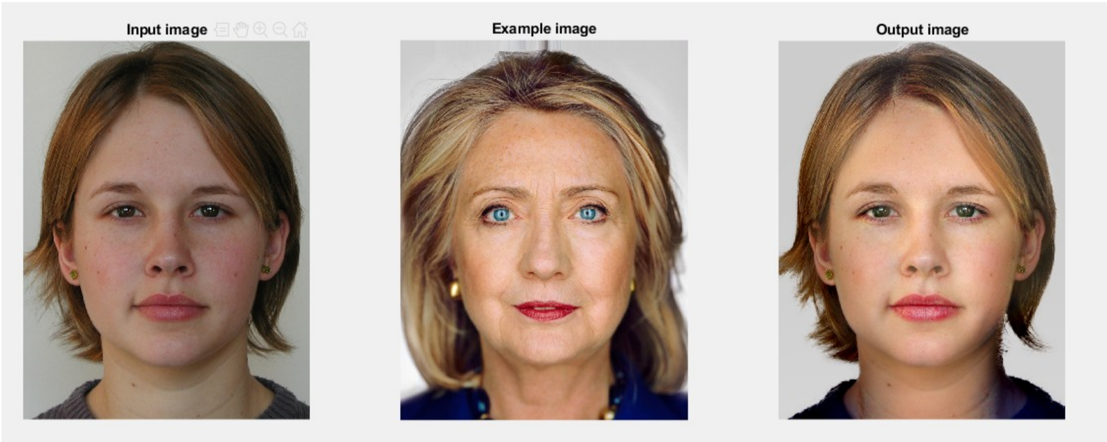
3-c)

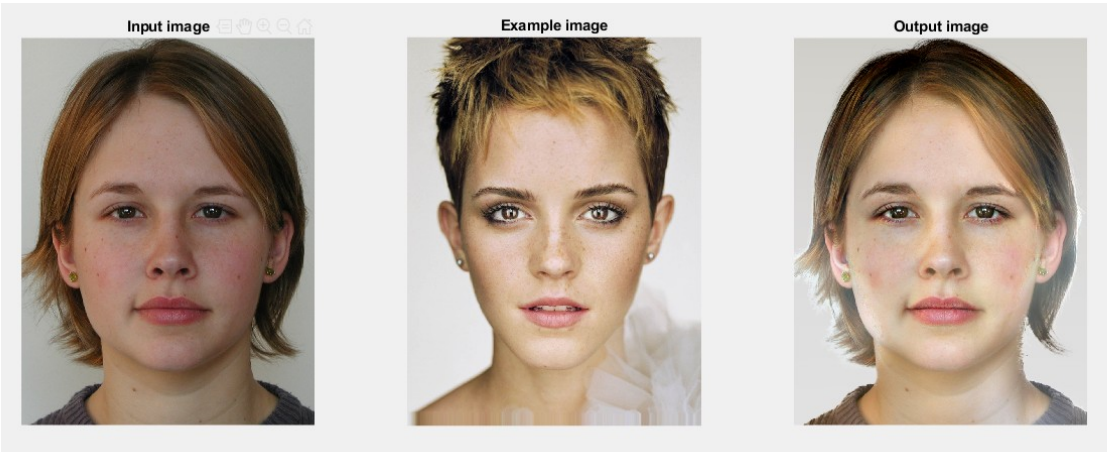


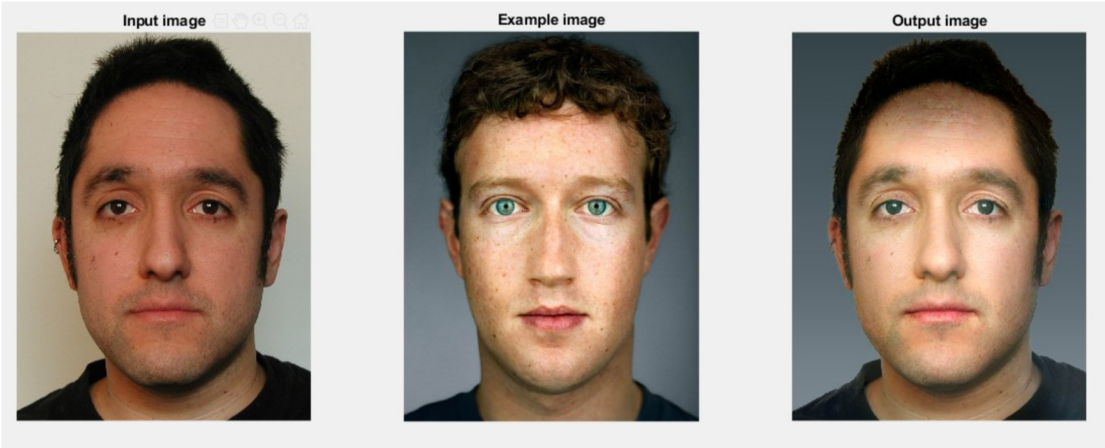
3-d , 3-f , 3-e)

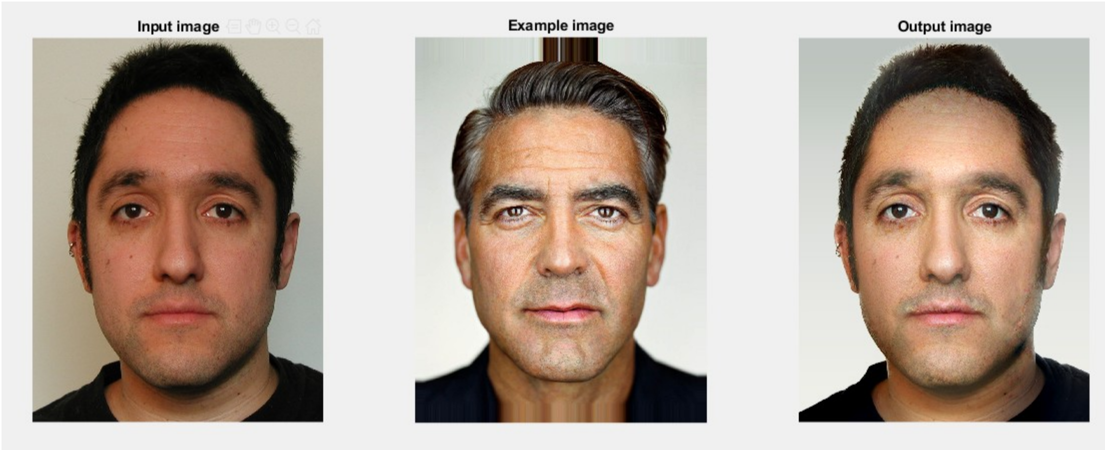

ex3-f
Input image
Example image
Output image

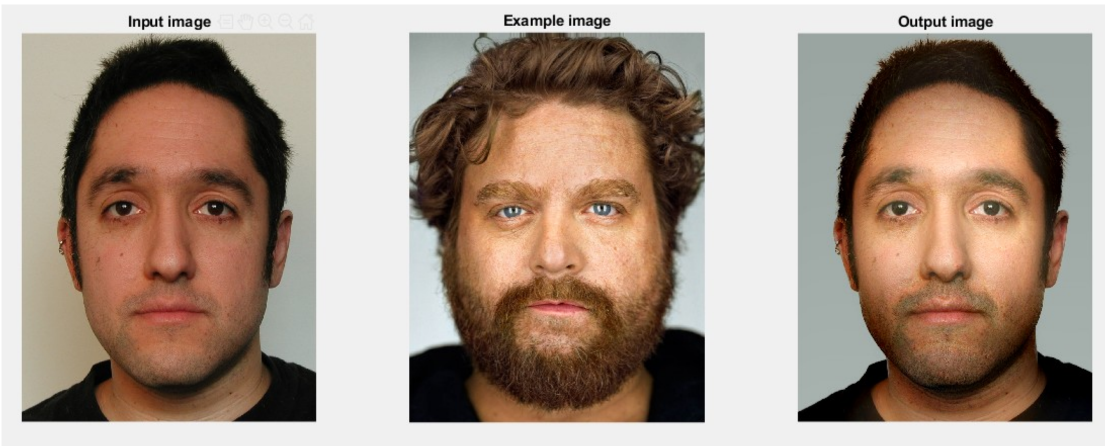
3-g)


Input image
Example image
Output image

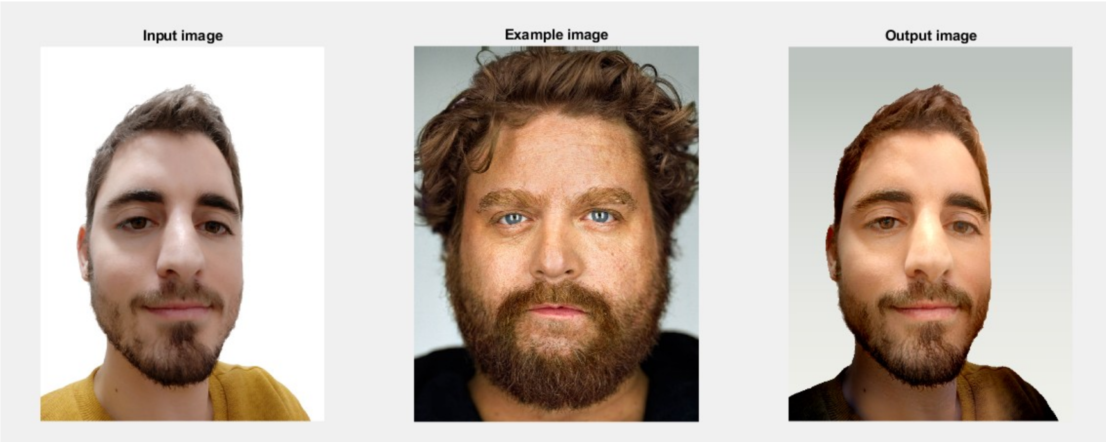

Input image
Example image
Output image


Input image
Example image
Output image


Input image
Example image
Output image


Input image
Example image
Output image

3-h)



3-i)

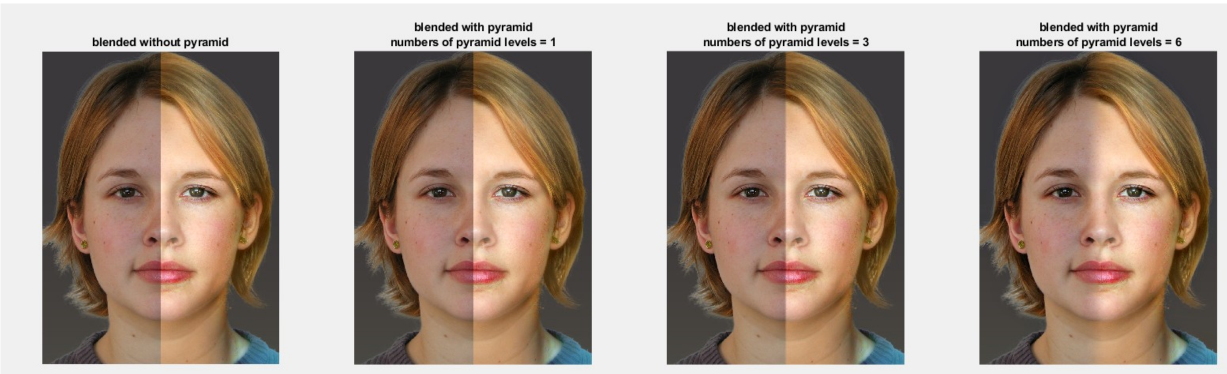


image blending with pyramids gives smoother blending. as much as the number of levels is bigger the stitching appear to be more natural.