4 Exercises

Exercise 1: Write a function to transform an input image based on the following transformation:

1. Rotation with the different degrees (45, -45, 120, -120, 180, 270)
2. Scaling images with scale = 2, 1/2
3. Translation image with $t_x = 50, t_y = 100$

For each case, the image result is saved with *jpg* format

Exercise 2: Write a function to perform perspective transformation for images on Google Drive of lab2.

1. With *book.png*, $P = \{(159.0, 125), (263, 183), (160, 336), (38, 234)\}$
2. With *paper1.png*, $P = \{(33, 97), (292, 111), (436, 457), (162, 577)\}$
3. With *ex2.png*, $P = \{(36, 62), (196, 19), (238, 102), (59, 163)\}$

Exercise 3: Write functions to convert an image from RGB color-space to gray with the following cases:

1. The lightness method averages the most prominent and least prominent colors: $(max(R, G, B) + min(R, G, B))/2$
2. The average method simply averages the values: $(R + G + B)/3$.
3. The luminosity method is a more sophisticated version of the average method: $0.21R + 0.72G + 0.07B$

Exercise 4: Write a program to compute the negative image of each gray image which converted in the previous exercise.

Exercise 5: Write functions to convert an image from RGB color-space to CMY color-space and reverse with the following equation:

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 255 \\ 255 \\ 255 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Exercise 6: Write functions to convert an image from RGB color-space to YCbCr and reverse which based on the equation below:

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.299R + 0.587G + 0.144B \\ 0.492(B - Y) \\ 0.877(R - Y) \end{bmatrix}$$

Exercise 7: Write function to conversion between RGB and YIQ with the following equations

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.274 & -0.322 \\ 0.211 & -0.523 & 0.311 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.0 & 0.956 & 0.621 \\ 1.0 & -0.272 & -0.649 \\ 1.0 & -1.106 & 1.703 \end{bmatrix} \begin{bmatrix} Y \\ I \\ Q \end{bmatrix}$$

Exercise 8: Write functions to compare two images according to the following:

- **Mean Squared Error (MSE)** :
  $MSE = \frac{1}{m.n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [f(x,y) - g(x,y)]^2$, a value of 0 for MSE indicates perfect similarity
- **Peak Signal to Noise Ratio (PSNR)**:
  $$PSNR = 10log_{10} \frac{M^2}{MSE}$$
  where M is the maximum of the intensity values in image.

Exercise 9: Write a program to evaluate the results between the transform images and the original image from the previous exercises.

1. Using MSE
2. Using PSNR

Exercise 10: Write a simple digital image processing program which performs the previous tasks (Exercise 1 - 9). It notes that the arguments are typed on command line by using **argparse** library

Hint:
```
import argparse
parser = argparse.ArgumentParser()
parser.add_argument('square', help='display a square of a given number',
type=int)
args = parser.parse_args()
print(args.square**2)
```

5 References

1. Reference 1
2. Reference 2
3. Reference 3
4. Reference 4
5. Reference 5