

## תרגיל בית - עיבוד תמונה

1. כתבו פונקציה בשם `load_image` המקבלת את מיקום תמונת הצבע והופכת את התמונה ל `np.array` - כלומר למערך, הפונקציה תחזיר את המערך.
2. כתבו פונקציה בשם `edge_detection` אשר מקבלת מערך המייצג תמונה (המערך שהתקבל מהפונקציה הקודמת) ומחזירה מערך חדש בעל אותו האורך והרוחב המדגיש את הקצוות בתמונה.  
לפי השלבים הבאים:

- צריך להפוך את המערך המכיל שלושה ערוצי צבע לתמונה אפורה המכילה ערוץ צבע אחד. יש לבצע זאת על-ידי מיצוע של שלושת ערכי הצבע עבור כל פיקסל.
- לאחר מכן, בנו פילטר (מערך דו-מימדי) שמוצא את השינויים בערכי הפיקסל בכיוון האנכי:

$$kernelY = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}$$

בנו פילטר שמוצא את השינויים בערכי הפיקסל בכיוון האופקי:

$$kernelX = \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix}$$

- הפעילו את כל אחד מהפילטרים על התמונה תוך שימוש בפונקציית `convolve2D` הנמצאת בספרייה `scipy.ndimage` – יש לבצע את הפעולה הבאה פעמיים: פעם עבור כל פילטר. יש לבצע את הקונבולוציה בעזרת `padding = 0` והתמונה שיצרתם לאחר הקונבולוציה צריכה להיות באותו אורך ורוחב כמו התמונה המקורית. אתם יכולים להיעזר בדוקומנטציה של הפונקציה:  
<https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.convolve2d.html>  
תקבל שני מערכים חדשים עסור כל אחד מהפילטרים, תוכלו לקרוא להם `edgeY` ו-`edgeX`.
- הפונקציה תחזיר מערך חדש שלוקח בחשבון את הקצוות בשני הכיוונים על-ידי הנוסחה:

$$edgeMAG = \sqrt{edgeX^2 + edgeY^2}$$

3. עכשיו נבנה את תמונות הקצוות שלנו:
- הורידו תמונת צבע בעזרת הפונקציה `load_image` שכתבתם, ניתן לבחור כל תמונת צבע שנמצאת במחשב שלכם. בתרגיל זה ניצור תמונה המבוססת על תמונה קיימת ומדגישה את הקצוות (`edges`) בתמונה.
  - מסכו את הרעשים בעזרת פילטר חציוני למשל כך:

```
from skimage.filters import median
from skimage.morphology import ball

clean_image = median(image, ball(3))
```

אתם יכולים לשחק ולשנות את ערך הפילטר 3 לערך אחר ולהציג את התמונה הממוסכת.

- הריצו את התמונה הנקייה מרעשים בפונקציה `edge_detection` שכתבתם.
  - הפכו את המערך שקיבלתם (*edgeMag*) מהפונקציה הקודמת לבינארי (אפס ואחד או True | False) על-ידי בחירת ערך סף. מומלץ להסתכל על ההיסטוגרמה של התמונה כפי שעשינו בתרגול בשביל לבחור את ערך הסף.
  - הדפיסו את התמונה שקבלתם ושימרו אותה.
4. העלו את התמונה המקורית ותמונת הקצוות שקבלתם לgithub.
5. עדכנו את הפונקציות שיצרתם `load_image` | `edge_detection` בקובץ `image_utils.py` בתיקיית github.
6. עדכנו את הקובץ `main.py` לפי הפעולות שביצעתם על מנת ליצור את תמונת הקצוות שלכם בתיקיית github.