ממ"ן 11

מגיש: נועם שדה

תאריך: 24/11/2022

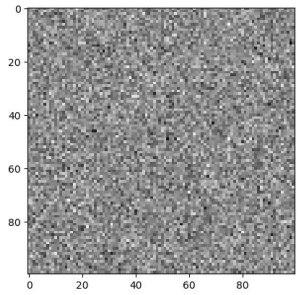
שאלות 1,2 נמצאות בקובץ main.py

לפני הרצת התוכנית יש להתקין את החבילות הנמצאות בקובץ requirements.txt
(pip install -r requirements.txt)
בנוסף יש לוודא שהקובץ LoG_filter באותה תיקייה וגם כל התמונות.

שאלה 1:

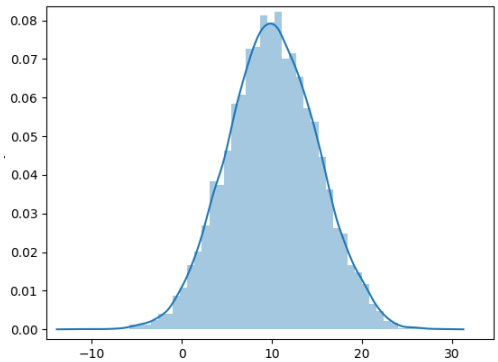
סעיף א':

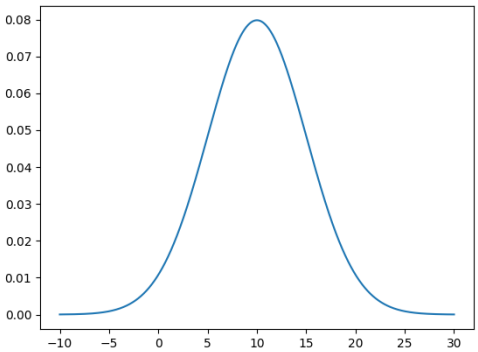
יצרתי מטריצה עם הפרמטרים mean=10, stdev=5, size = 100x100 בעזרת numpy והצגתי אותה כתמונת רמות אפור דו ממדית בעזרת matplotlib.



סעיף ב':

הצגתי היסטוגרמה של המטריצה מסעיף א' השווה לפונקציית הפילוג המתאימה לה.





ניתן לראות שהיא מתאימה לפונקציית הפילוג המתאימה לה.

סעיף ג':

הצגתי תמונה של מסי צבעונית ותמונת רמות אפור

סעיף ד':

ביצעתי canny edge detector על התמונה בעזרת opencv Canny. קיבלתי תוצאות יותר טובות על התמונה הצבעונית ולכן המשכתי איתה. ביצעתי canny edge detector עם שלושה פרמטרים שונים:

(100,200),(150,250),(200,300)

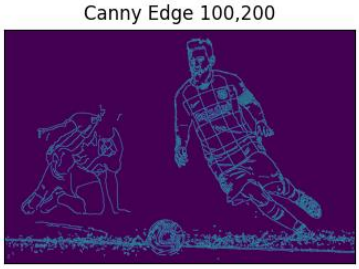
כאשר בסט הפרמטרים (100,200) רואים הרבה קווים

ובסט הפרמטרים (200,300) רואים פחות קווים אך התמונה עדיין ברורה.

ניתן לראות שכאשר סף המינימום נמוך יש הרבה רעשים בתמונה וקשה יותר לזהות edges.

Original Image

Canny Edge 100,200



Canny Edge 150,250

Canny Edge 200,300

סעיף ה':

חישבתי Harris corners בעזרת HarrisCorners בספריה opencv. ניסיתי שני סטים של פרמטרים וצבעתי את הנקודות על התמונה באדום.

סט 1: (2,3,0.04) בחרתי פרמטרים אלה לפי הדוקומנטציה של opencv.

סט 2: (3,5,0.07) בחרתי לפי ניסוי וטעיה.

נשים לב שבסט 1 רואים בבירור את הקצוות לעומת סט 2 שיש הרבה יותר נקודות והם פחות ברורות.

הסבר על הפרמטרים לפי האתר:

OpenCV has the function **cv.cornerHarris()** for this purpose. Its arguments are:

- **img** - Input image. It should be grayscale and float32 type.
- **blockSize** - It is the size of neighbourhood considered for corner detection
- **ksize** - Aperture parameter of the Sobel derivative used.
- **k** - Harris detector free parameter in the equation.

(2,3,0.04)

(3,5,0.07)

שאלה 2:

*הערה: יש להמתין בערך 50 שניות לריצת התוכנית.

סעיף א':

האלגוריתם שכתבתי רץ בלולאה על כל התמונות בתיקייה images ועבדתי לפי השלבים המתוארים בשאלה בממ"ן.

1. בניית הפירמידה: יצרתי סט של פילטרים עם הפרמטרים לפי ההנחיות. הגדרתי שיהיו 15 פירמידות(ככה נמצא הכי הרבה נקודות) ולכן בסט הפילטרים יש 15 סקיילים של פילטרים. הגדרתי את התמונות שיהיו בגודל 512x512 ויצרתי את הפירמידות עם convolve2d .

2. Non-maximum suppression: הגדרתי את הפרמטר suppression_diameter להיות החציון של הסקיילים ובעזרתו חיפשתי נקודות מקסימום לוקאליים במערך התלת ממדי.

3. הצגתי את הנקודות את התמונה.

הסבר לנרמול הפילטר ע"י מכפלה ב $sigma^2$ בסעיף 2:

The filter response decreases as sigma increases. For each gaussian derivative, the response to shock decreases by a factor of sigma. Since Laplacian is second derivative, we get $sigma^2$.