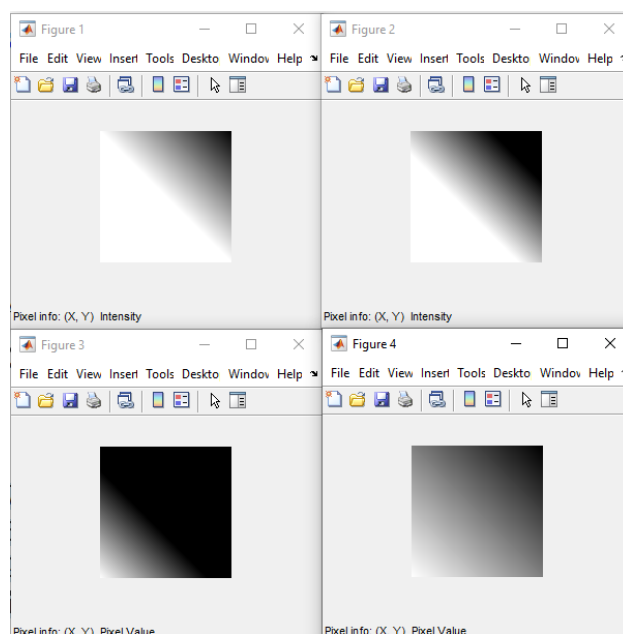
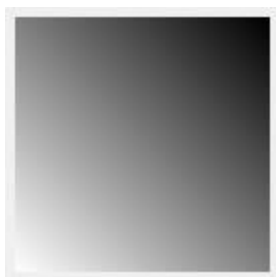


דוח מכין 1 מעבדה בעיבוד תמונה – 061

נתנאל רוטשילד 204937841

שבי שבתן 305340713

תרגיל (1) הרצה ראשית של הקוד



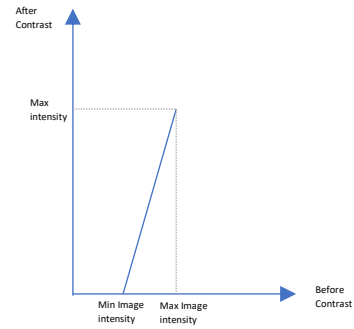
שורות קוד הבאים.

1. ב MATLAB התמונות מיוצגות בצורה של מטריצות.
BINARY - הינו מטריצה ("תמונה") בעל 2 ערכים שונים
COLORED - היא תמונה בעלת 3 מימדים, המימדים הראשונים הם x, y והמימד השלישי הוא הערוץ (Channel) כאשר כאן יש 3 ערוצים שהם אדום, ירוק, כחול
GREYSCALE - הינו יצוג של תמונה בערוץ בודד, כאשר לטווח ערכים באפשריים נעה כתלות בטיפוס.
INDEXED - בדומה ל COLORED גם מייצג תמונה בכמה צבעים, רק שזוהי מטריצה בעלת 2 מימדים x, y וכל איבר במטריצה בעלת 3 ערכים בצבעים אדום, ירוק, כחול
2. תחום דינאמי מתייחס לטווח שבין הערך הקטן ביותר בתמונה לבין הערך הגדול ביותר שפיקסל יכול לקבל. ערך זה קובע לנו את טווח הגוונים האפשריים. אפשר לראות בתמונות שהוצאנו כי הטווחי מעבר בין הקצוות מהיר (קטן) יותר, לעומת זאת בתמונה 4 השינוי נפרס כמעט לאורך כל התמונה.
3. שימוש ב- [] מסמן טווח דינאמי מקסימלי, כאן טווח זה הוא [0,255].

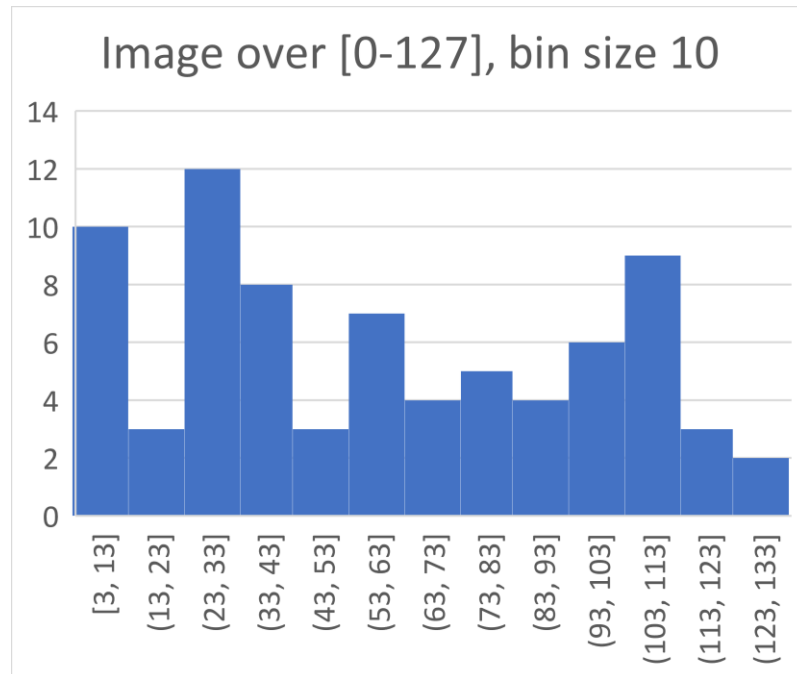
4. Impixelinfo מאפשר לנו לקבל מידע על pixel שאנחנו מצביעים עליו, שימושים יכול להיות קבלת ערך מסוים לבחירת טווח (סף) ערכים שאנחנו מעוניינים להציג

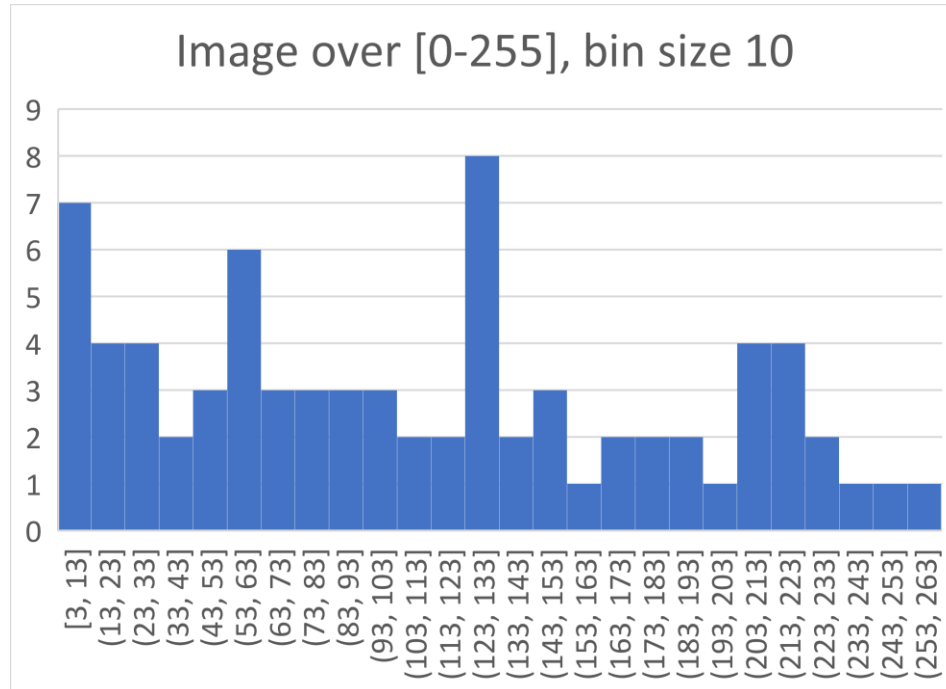
תרגיל (2)

1. היסטוגרמה של תמונה היא שכיחות ערכים בתמונה. אפשר להבין מתוך ההיסטוגרמה מהי הטווח הערכים שיש בתמונה, ואם יש טווחים יותר נפוצים מאחרים
2. בעזרת מתיחה להיסטוגרמה אנחנו מגדילים את הטווח הדינאמי של התמונה
3. עבור ערכים של התמונה המקורית נפרוס אותם על טווח גדול יותר



4.



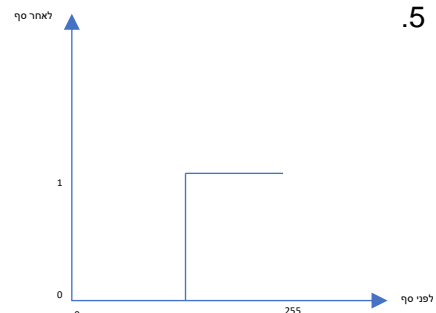


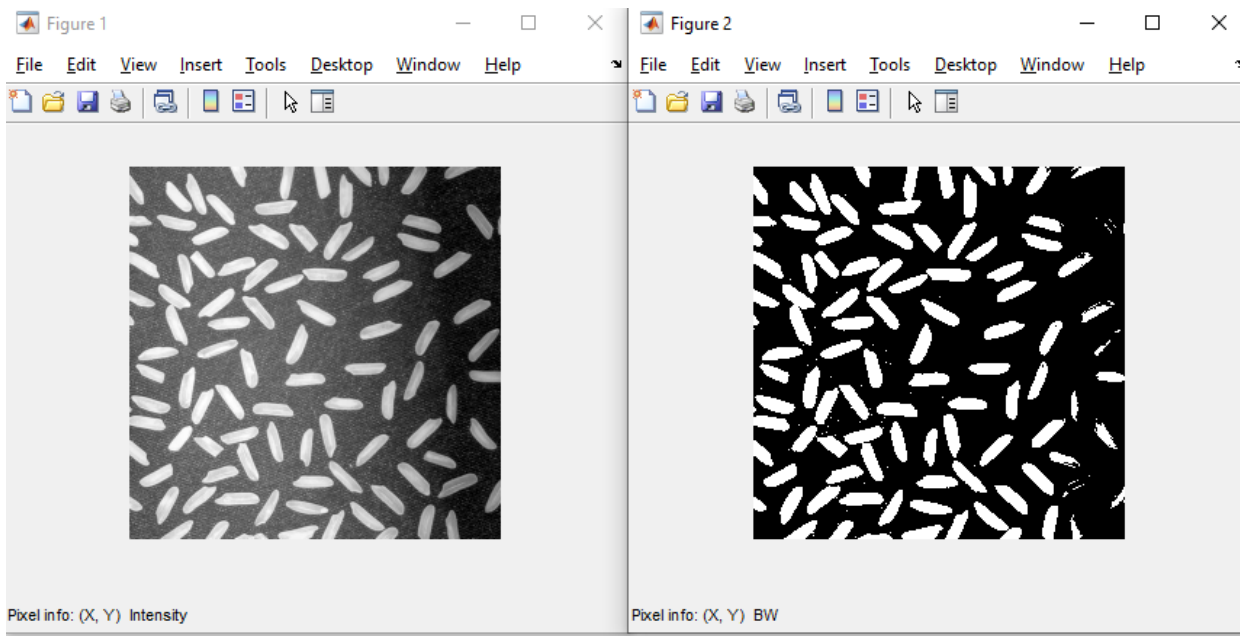
5. שוויון היסטוגרמה היא פעולה שלא דורשת התערבות מצד המשתמש. בפעולה הזו כל הרעיון הוא ל"פזר" את הערכים לאורך כל ההיסטוגרמה. נעשה על ידי ממוצע משוקלל של הערכים. בניגוד למתיחות של ההיסטוגרמה שהיא דורשת זיהוי טווח על ידי המשתמש אותו "נפרוס" לאורך ההיסטוגרמה.
6. פעולת המתיחה אינה לינארית, לדוגמא עבור מתיחות בשני טווחים שונים ואז סכימה ביניהם תתן תוצאה שונה מסכימה בין 2 היסטוגרמות ואז מתיחה עבור הגדול מבין השנים שעשינו בדוגמא לפני.

תרגיל 3)

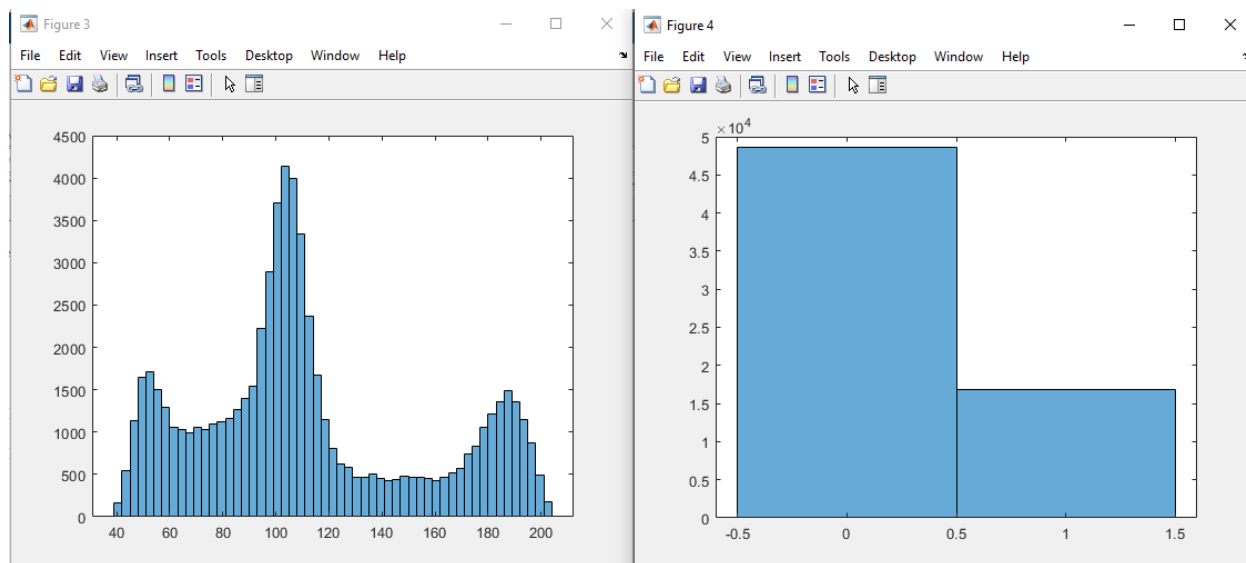
1. קישוריות היא מציאת שכנים של פיקסל בתמונה, שכן עבור פיקסל $[i, j]$ מוגדר כ- $[i-1, j-1], [i, j-1], [i+1, j-1], [i-1, j], [i, j], [i+1, j], [i-1, j+1], [i, j+1], [i+1, j+1]$ עבור הגדרה 4 connectivity ועבור 8 connectivity נוסף גם האלכסונים. שכן הוא פיקסל בעל ערך זהה לפיקסל אתחול.
2. פונקציה `bwconncomp` מקבלת תמונה בינארית יחד עם סוג הקישוריות שנרצה. מחזיר את גודל התמונה ומספר האובייקטים שנמצאו בתמונה.
3. פונקציה `labelmatrix` מקבלת את הפלט של `bwconncomp` ומחזירה תיוג לכל פיקסל לאיזה אובייקט הוא משויך.
4. חיתוך בסף הופכת קלט (תמונה) לתמונה בינארית, כאשר ה"סף" לאיזה ערך לתת לפיקסל בתמונה המקורית. הפעולה אינה לינארית, סכימה בין 2 תמונות שונות ואז הפעלת סף עלולה לתת תמונה שונה מביצוע סף על כל אחת ואז חיבור (הכפלה כי מדובר ב1/0) ביניהם. הוא שימושי להכנת תמונה לקראת הפונקציות קישוריות.

5.





לדוגמא:



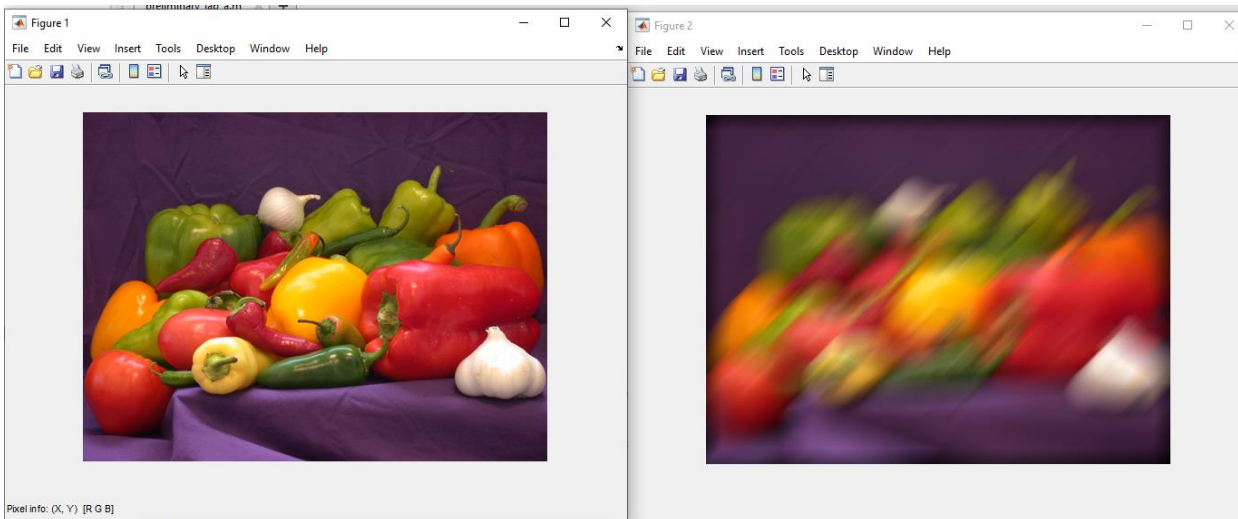
6. ניתן לבדוד את הרקע עלידי בשלושה שלבים ראשית נגדיר את הסף להיות ϵ גדול מערך הרקע ונשמור את התמונה הבינארית המתקבלת. שלב שני נגדיר את ערך הסף להיות ϵ קטן מערך הרקע, ומיד נהפוך אותו (כל 1 יהפוך ל0 וכל 0 ל1) שלב שלישי נסכום עם התמונה הראשונה שהוצאנו, ככה הצלחנו לתפוס את על מה שלא רקע.

7. זיהוי פנים בתמונות וזיהוי לוחית רישוי בכביש 6.

תרגיל 4)

1. רעש מסוג Poisson הינו רעש לאורך כל התמונה כאשר לכל פיקסל מתווסף ערך בהתפלגות Poisson. Salt & Pepper הינו בחירת פיקסלים באופן אקראי כך שערכם ישתנה באופן קיצוני מערכם המקורי.

- Speckle הינו רעש שנובע כתוצאה של התנגשות בין החזרי אור. ערך של פיקסל מורעשת היא $I + n \cdot I$ כאשר I היא הערך המקורי ו- n בעל התפלגות אחידה עם תוחלת 0.
- מקרה שחלק מהקולטנים לא תקינים יכול להכניס רעש לתמונה הסופית. במידה וחלק מהחיישנים לא תקינים נקבל רעש מסוג Salt & Pepper לדוגמא
 - fspecial פונקציה שמקבלת סוג פילטר ויוצרת פילטר דו מימדי מתאים.
 - impfilter פונקציה המקבלת מערך בעל n מימדים וfilter בעל n מימדים ומעבירה את הפילטר על המערך.
 - ניצר פילטר בעזרת fspecial אותו ניתן להעביר בעזרת impfilter
 - מסנן חציון הינו מסנן המשבץ לכל פיקסל את הערך בעל החציון מתוך החלון של המסנן. יתרון במסנן זה שנשמרים ערכים שהיו קיימים בתמונה המקורית (יתכן והוזזו), גם תוכל לשמור על edges בצורה טובה. חסרון הוא שאם מדובר בתמונה עם הרבה ערכים קיצוניים, הערכים המנוגדים יבלעו.
 - מסנן ממוצע הינו מסנן שמחליק חלון על התמונה כאשר לכל פיקסל נכנס הממוצע של החלון. יתרון של חלון זה ערכים שונים מאוד מהשכנים שלו עדיין יראו שונים. חסרון הוא שאם רוב הפקסלים באותו טווח, יתכן והם ימרחו לאותו ערך ולא נוכל להבדיל ביניהם.
 - Circular כאשר נעביר את המסנן על התמונה מניחים שערכים מחוץ לתמונה הינו ערך בקצה השני של התמונה (מחזורי)
 - Replicate מניח כי מחוץ לגבולות התמונה נקבל פשוט העתק של גבול התמונה
 - Deconvlucy היא פונקציה המבצעת שיחזור של תמונה שעבר סינון (טשטוש) תחת הנחות מסוימות. Psf הינו פונקציה המתארת את הגרעין של התמונה.
 -



תרגיל 5)

- קוונטיזציה זו תהליך של המרת קלט לקבוצות קטנות יותר. לדוגמא איחסון של מידע רציף (גלי אור) לערכים סופיים. או לחילופין דחיסה של תמונה לתמונה קטנה יותר.
- פרמטר levels הינו ווקטור של פרמטרים כאשר במידה וערך הקלט קטן מהמינימום בווקטור, הקלט יקבל ערך 1. אם הוא גדול מהערך המקסימלי בווקטור, הקלט יקבל את הערך $N+1$. אחרת נמצא את הטווח בו נופל הקלט נגיד בין $levels(i-1)$ ל- $levels(i)$ אז הקלט יקבל את הערך i .
- פרמטר values מחזירה את הערך הרצוי בכל אחת משלושת האפשרויות שהוגדר מקודם.
- שגיאה MSE הינו שגיאה ריבועית בין שני משתנים (מטריצות, תמונות) בעלי אותם מימדים וגדלים. כאשר הוא נתון באופן הבא: $MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i,j) - K(i,j)]^2$
- יצא לנו $MSE = 589.6512$ המסקנה שלנו כי התמונות לא דומות

5. MSE המינימלי שניתן לקבל הוא אפשרת עבור השוואה בין 2 ערכים זהים, והערך המקסימלי לא מוגדר ותלוי בטווח ערכים שניתן לקבל עבור כל פיקסל, עבור תמונה נתונה נקבל את MSE הגדול ביותר עבור ערכים קיצוניים רחוקים מהתמונה המקורית.
6. התמרת DCT עושה שימוש ב \cosine כפונקציית בסיס, בניגוד להתמרת פורייה שעושה שימוש ב $exponents$. כאשר נעשה שימוש בהתמרת DCT בצהליך דחיסת תמונות, רוב המידע יהיה בתחומים הנמוכים, מכאן שניתן לייצג את רוב המידע בתמונה בעזרת מעט פונקציות \cosine ואת רוב המקדמים נוכל לאפס. זהו הבסיס לשיטת דחיסה מסוג $JPEG$.
7. זהו דומה להתמרה של פונקציית דלתא, סביר להניח כי נקבל פס לאורך התמונה (אנכי או מאוזן, תלוי איך נבצע את ההתמרה). לא בדקנו, מניחים

שאלה 6:

1. תמונת שפות הינה תמונה בינארית שבה פיקסל לבן מסמל שערך הנגזרת של התמונה המקורית בפיקסל זה גבוה מסף מסוים. ערך נגזרת גבוה מסמל שפות כיוון ששפות מאופיינות בהפרשים חדים בין פיקסלים.

2. האופרטורים לגילוי שפות הינם :

- SOBEL-שיטה זו מוצאת את תמונת הנגזרות באמצעות שימוש באופרטור שמקרב נגזרת של אות דיסקרטי בגודל שלוש על שלוש מהצורה הבאה:

$$G = \sqrt{G_x^2 + G_y^2}$$

$$G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} * A \quad G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * A$$

כאשר A הינה התמונה שמחפשים את השפות שלה.

- CANNY-שיטה זו משתמשת ב-2 ערכי סף. סף אחד למציאת שפות חזקות וסף שני למציאת שפות חלשות. השפות החלשות ייכללו בתמונת השפות במידה שהן מחוברות לשפות חזקות.
- LAPLACIAN OF GAUSSIAN-שיטה זו מוצאת את תמונת הנגזרות באמצעות קירוב לאופרטור גזירה של לפלסיאן. 2. אופרטורים רלוונטיים לשיטה זאת הם :

$$D_{xy}^2 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad D_{xy}^2 = \begin{bmatrix} 0.25 & 0.5 & 0.25 \\ 0.5 & -3 & 0.5 \\ 0.25 & 0.5 & 0.25 \end{bmatrix}$$

הפרמטרים אשר האופרטורים מקבלים הם ספים וכיווני גזירה.

3. ההבדלים המהותיים בין האופרטורים ותוצאות פעולתם באים לידי ביטוי ב:

- רגישות תמונת השפות לרעשים.
- אופן חישוב השפות.
- עומס חישובי בעת מציאת השפות.

לדוגמא אופרטור SOBEL פשוט חישובית ורגיש לרעשים.

4. שתי סיבות שונות להיווצרות שפות בתמונה הן :

- מעברים בין אובייקטים לאובייקטים אחרים/רקע כלשהו-מעברים אלו ייגרמו להיווצרות נגזרות חזקות שיזוהו כשפות בתמונה.
- אובייקטים שאינם חלקים-אובייקט שאינו חלק ומכיל שינויים חדים בין איזורים שונים בו יכיל שפות אשר מפרידות בין אזורים השונים.

5. שתי בעיות בתהליך גילוי השפות:

- אי זיהוי שפות-איזורים של שפות בתמונה אשר לא מזוהים כשפות. הדבר עלול להיגרם כתוצאה מהגדרת ערך סף גבוה מדי, תמונה רועשת וסינון שגרם לטשטוש של שפות.
- זיהוי שווא של שפות-איזורים שאינם שפות בתמונה שמזוהים כשפות. הדבר נגרם כתוצאה מאיזורים בעלי נגזרות חזקות מספיק שמוגרות כשפות .