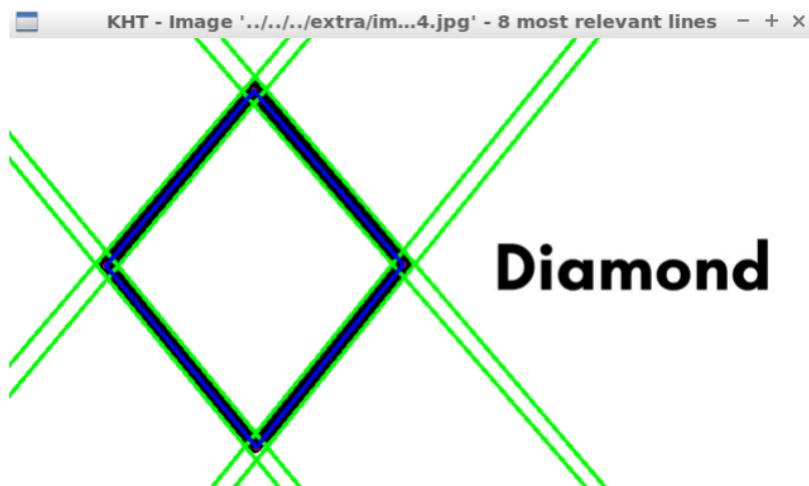
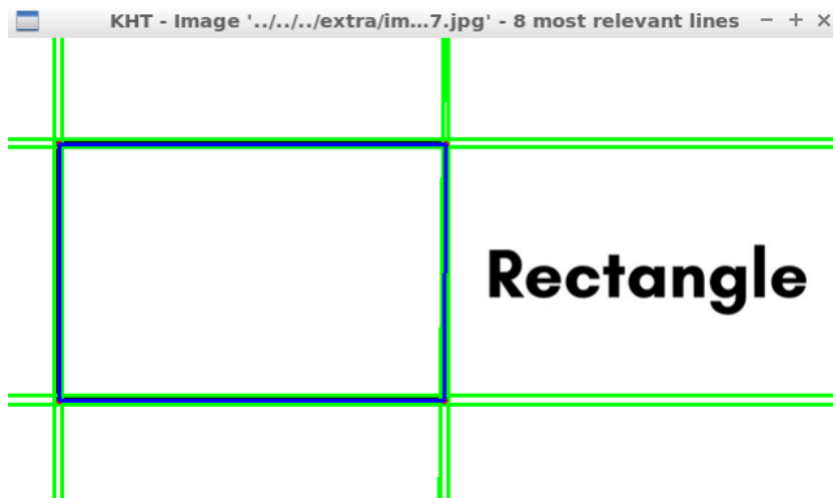
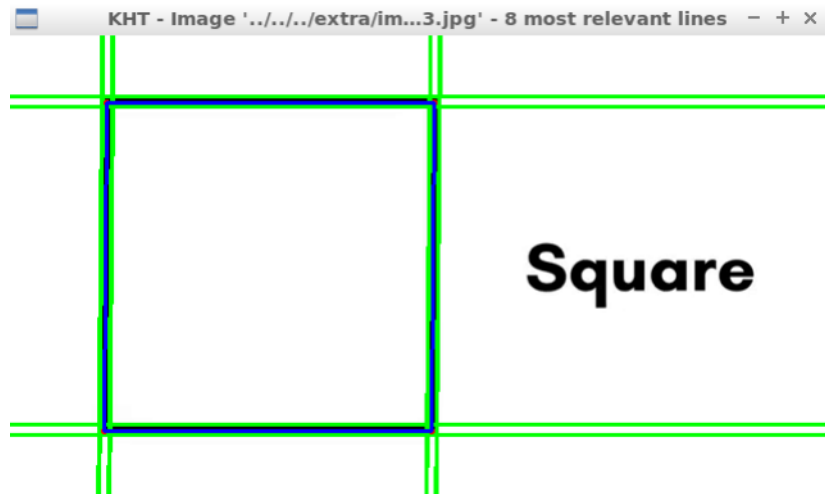


תיאור הפרויקט

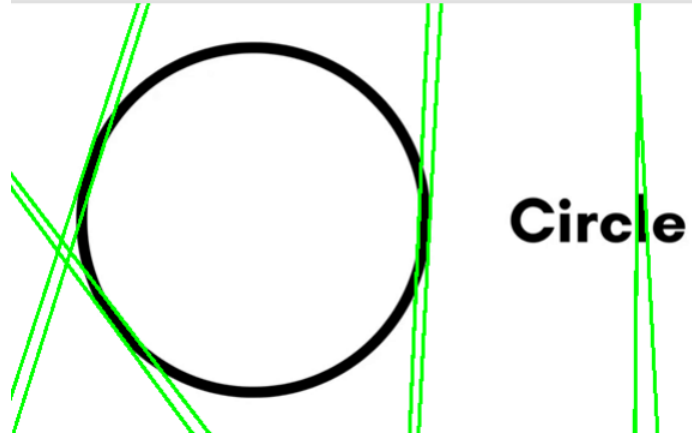
- בפרויקט זה, אנו מעוניינים לזהות מרובעים המופיעים במהלך קטע וידיאו נבחר.
- תחילה, אנו קולטים קטע וידאו וממירים אותו למערך של 10 תמונות הנבחרת באופן אקראי מקטע הוידאו.
- לאחר מכן, אנו מוצאים נקודות קצה עבור כל תמונה באמצעות האלגוריתם `canny`.
- בהינתן נקודות קצה, נשתמש באלגוריתם הידוע `hough-transform` המשודרג על מנת לזהות את הקווים שבתמונה (הפונקציה `run_kht`).
- כעת נרצה למצוא את נקודות החיתוך בין הקווים שהתקבלו כדי לנסות להרכיב מרובע מנקודות חיתוך אלו (הפונקציה `findIntersections`).
- במידה ונקבל מספר רב של נקודות חיתוך, נרצה לצמצם את מספר זה באמצעות האלגוריתם הידוע `k-means-clustering` המספק 4 נקודות מרכזיות בתמונה.
- בסופו של דבר, נבדוק האם 4 הנקודות הנתונות מהוות מרובע תקין באמצעות הפונקציה `findQuadrilaterals` הכולל בתוכו מספר תנאים.

תוצאות הפרויקט

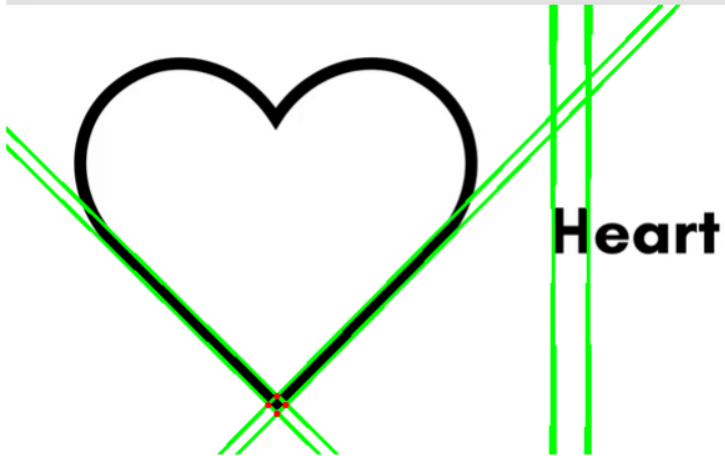
- התמונות הבאות הן חלק מקטע הוידאו הנבחר.
- צבע ירוק – הקווים המתקבלים מהאלגוריתם המשודרג של `hough-transform`.
- צבע אדום – נקודות החיתוך המרכזיות המתקבלות מהאלגוריתם `k-means-clustering`.
- צבע כחול – המרובע התקין המתקבל מהפונקציה `findQuadrilaterals`.



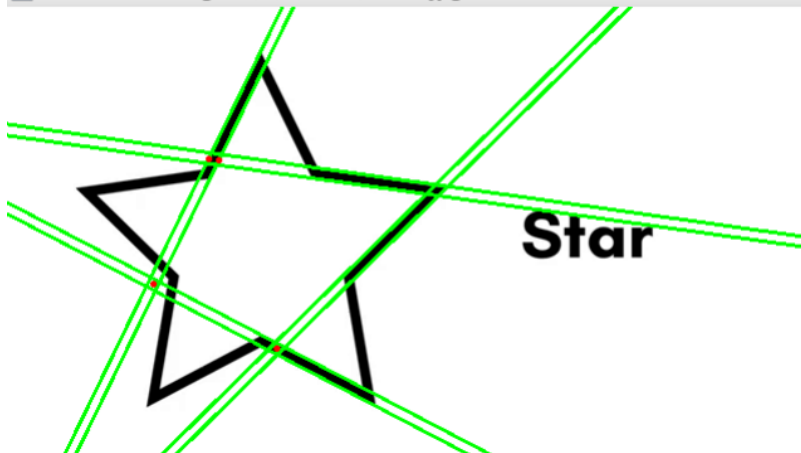
KHT - Image '../..../extra/im...6.jpg' - 8 most relevant lines - + x



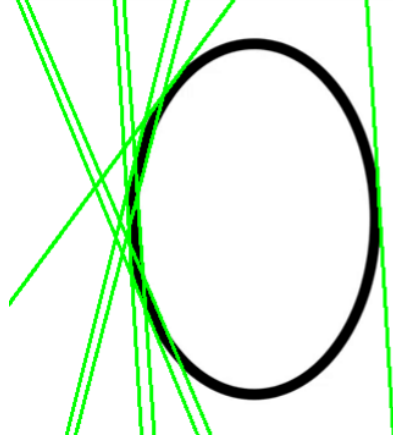
KHT - Image '../..../extra/im...8.jpg' - 8 most relevant lines - + x



KHT - Image '../..../extra/im...7.jpg' - 8 most relevant lines - + x

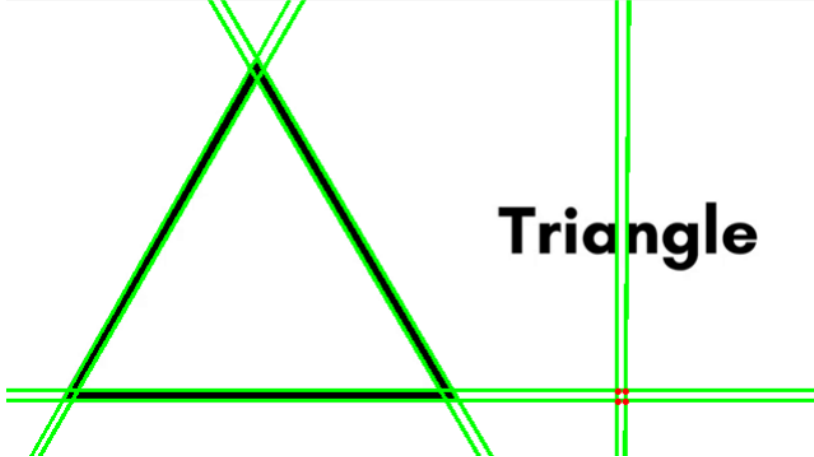


KHT - Image '../..../extra/im...2.jpg' - 8 most relevant lines - + x



Oval

KHT - Image '../..../extra/im...4.jpg' - 8 most relevant lines - + x



Triangle