

## **פרויקט מסכם עיבוד תמונה – "מלר הרקפות"**

מרצה: פרופ' תמי ריקלין רביב

מתרגלות: שירה כרמי, יעל אלקין

### **מגישים:**

דורון גוזנברג 318183589

מאור בראון 316119767

עומר הורוביץ 322587395

מתן גבאי 209268689



## תוכן עניינים

3 .....	תקציר הפרויקט :
4 .....	תיאור הפרויקט .....
6 .....	תיאור האלגוריתמים .....
7 .....	זיהוי הcadors: .....
12 .....	אלגוריתם לזיהוי הקפצת כדורים: .....
14 .....	זיהוי פגעה ברכפה: .....
20 .....	אלגוריתם זיהוי נגיעה ביד .....
26 .....	אנליזת התוצאות - בחינת ביצועי אלגוריתם זיהוי הרצפה .....
32 .....	מסקנות: .....
33 .....	סיכום: .....
34 .....	ביבליוגרפיה: .....

## תקציר הפרויקט :

משחק הcador גול הוא בין המשחקים הפופולריים בעולם. כאשר מיליון ילדים ברחבי העולם מקדישים את מרבית זמנם בביולי במשחקי הcador גול. אחת הדרכים להשתפר במשחק היא להשתפר בשליטה בcador על ידי שליטה במינונות הקפוץ. ילדים רבים מתקשים בספירת הקפוץ באופן סימולטני עם ביצוע הקפוץ. לכן יצרנו תחרות שתדרבן את השחקנים להשתפר פלאים במשחק הקפוץ תוך כדי הנגשת הסירה למשתמש בצורה הנוחה ביותר. מלך הקפוץ היא תחרות הקפוץ כדור כדור גול. ראשית השחקן מכניס את שמו המלא למערכת ולאחר מכן מקפץ אתcador עד שנופל. פסילה מוגדרת להיות אחת משלושת האפשרויות הבאות:

- א.cador נופל לריצה.
- ב.cador יוצא מתחומי המשחק (טוווח הראייה של המצלמה)
- ג. השחקן נוגע בcador בעזרת היד.

מספר הקפוץ נספר באופן אוטומטי, כאשר השחקן נפסל באחת הדרכים המתוארות לעיל, האלגוריתם פועל את השחקן ומציג את מספר הקפוץ שצבר, מתאר את סיבת הפסילה ואת מיקומו בטבלה ביחס לשאר המשתתפים.



## **תיאור הפרויקט**

### תיאור המערכת:

לאחר הכנסת שם השחקן, מנהל המשחק מבקש מהשחקן לעמוד מול המצלמה במרכז הפרויקט כאשר ידיו על מותני ומנהל המשחק מסמן את ידיו של השחקן, ומפעיל את הספירה לאחר שבסופה מגיעה הזמןket המשחק. השחקן מופיע את הcador שלו נזהה באלגוריתם, נזהה הקפיצה חוקית ונספר את מספר הקפיצות החוקיות עד אשר נפסל לאור אחת מן הסיבות הבאות: הcador נופל לרצפה, הcador יוצא מתחום המשחק (טוויה הריאיה של המצלמה) או נגיעה היד של השחקן בcador. את הסיבות הללו אנו נזהה בזמן אמת ואשר השחקן נפסל האלגוריתם פוסל את השחקן ומציג את מספר הקפיצות שצבר, מתאר את סיבת הפסילה ואת מיקומו בטבלה ביחס לשאר המשתתפים.

### מיקום המצלמות ויעודן:

בפרויקט אנו משתמשים במכשיר אחד. המצלמה תמוקם מול השחקן כאשר הוא עומד מולה בתוך שטח המשחק שהוגדר לו. המצלמה תמוקם כך שבתוך הפרויקט יתפס כל גוף של השחקן ומעט שטח מעל ראשו, בនוסף הפרויקט יכלול חלק נכבד מן הרצפה לפני השחקן לצורך זיהוי פגיעה ברצפה. יעוד המצלמה הוא לקלוט את הקפיצות הcador ופסילת המשחק.

### מטרות:

מטרת העל של הפרויקט היא לספק פתרון יעיל ואמין לספירת הקפיצות כדור חוקיות בזמן אמת ובאופן רציף.

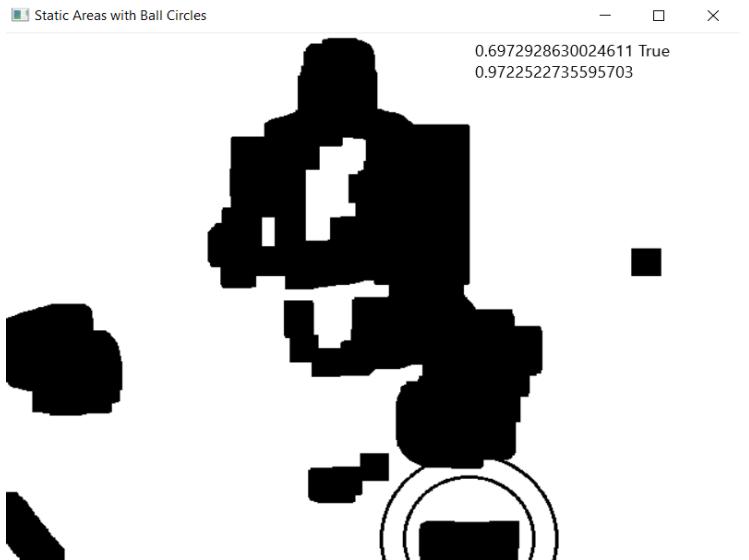
1. זיהוי הcador בכל פרויקט.
2. זיהוי הקפיצה חוקית.
3. זיהוי מיקום הידיים בתחילת המשחק.
4. מעקב אחרי ידי השחקן לאורך המשחק בזמן אמת.
5. ספירת הקפיצות
6. זיהוי פסילת המשחק באמצעות מהאפשרויות הבאות:
  - a. זיהוי מגע בין ידי השחקן לcador.
  - b. זיהוי נפילת הcador על הרצפה.
  - c. יציאת הcador מהפרויקט.
7. עיצוב המשחק, חווית המשתמש והציג הניקוד וסיבת הפסילה בזמן אמת.

## AILOZI/הנחות הפרויקט:

1. הcadור אדום.
2. השחקן אינו לובש בגדים אדומים.
3. הרקע אינו מכיל חלקי אדומים.
4. כמעט השחקן והcadור אין אובייקטים נוספים שזרים בפראיים.

## אתגרי הפרויקט:

1. למצוא אלגוריתם שיצילח לעקב אחר תנועות ידו של השחקן בדגש על תנועות חדשות (נראה כMRIיה בפראיים).
2. למצוא אלגוריתם שיזהה הקפיצה תקינה ויספור כל הקפיצה כהקפיצה יחידה ויקלוט את כל הקפיצות התקינות.
3. מציאת אלגוריתם לזיהוי נפילתcadור לרصفה שיתמודד בצורה מיטבית עם מקרה קצה שבוcadור נופל סמוך לרגלי השחקן (עלול להראות כהקפיצה תקינה).
4. זיהויcadור בכל פראיים ללא קשר למרחקו מן המצלמה ולהירותו.
5. התמודדות עם מצב קצה הקשורים למיקומים של השחקן ושלcadור ביחס למצלמה וביחס בין מיקום השחקן לכדור עצמו.
6. מציאת אלגוריתמים יעילים לשיפוט שיאפשרו הרצתה של המשחק בזמן אמת ללא פגיעה באיכות המשחק וחווית המשטמש.
7. אינטגרציה של כל חלק המשחק לכדי קוד אחד מתפקד.



בתמונה זו רואים תמונה הפרשים שבה מתקיים מצב שבוcadור נופל על הרצתה אך קרוב מדי לרגלי השחקן ולכן זהה כהקפיצה תקינה.

## תיאור האלגוריתמים

הקוד שלנו מכיל ארבעה אלגוריתמים עיקריים:

**אלגוריתם ראשון** הינו אלגוריתם שמטרתו לזהות את הcador, כאשר אין זיהוי של הcador נסיק כי הcador יצא מחוץ המשחק ולכן נפסול את השחקן.

**אלגוריתם שני** שמטרתו לזהות מתי התרחשה הקפיצה ובנוסף מנית הקפיצות השחקן.

**אלגוריתם שלישי** שמטרתו לזהות את ידיו של השחקן ולקוב אחר תנועת, ולפסול את השחקן במקרה של פגיעה ביד.

**אלגוריתם רביעי** שמטרתו לזהות האם התרחשה פגיעה של הcador ברצפה אשר טוביל לפסילת השחקן.

על מנת לשפר את חווית המשתמש של השחקן השתמשנו בתיקיית "Pygame". יצרנו מסך המציג ספירה לאחור של השחקן וכאשר הוא מגיע ל-10 הקפיצות מופיעה תמונה של זיקוקים עם קריית שמח "wow" בנוסף, כאשר השחקן נפסל הוצאנו כפלט על המסך הודעה של "game over" לצד כמהות הקפיצות שהשחקן הצליח להקפי וסיבת פסילתו בתוספת סאונד של הפסק. יחד עם זאת מימשנו באותה תיקייה טבלה שבה השחקן מכניס את שמו המלא על מנת להיכנס לתחנות ובסוף המשחק הטבלה מתעדכנת עם מיקום השחקן בטבלה.

Your score is: 56

ball out of frame

**זיהוי הcad:**

### **רקע ומוטיבציה:**

חלק עיקרי במשחק הקפוץ הוא לזהות היכן נמצא הcadור בכל פריים, על מנת שנוכל לעקוב אחר תנועתו ולזהות כל הקפוצה וכל פסילה.

**קלט האלגוריתם** – פריים ממצלמת הידאו.

**פלט האלגוריתם** – שיעור נקודת מרכז המסה של הcadור ורדיוו, בנוסף תחימת הcadור בכל פריים על ידי המעל שהתקבל מנקודת מרכז המעל והרדיוו.

### **מהלך האלגוריתם:**

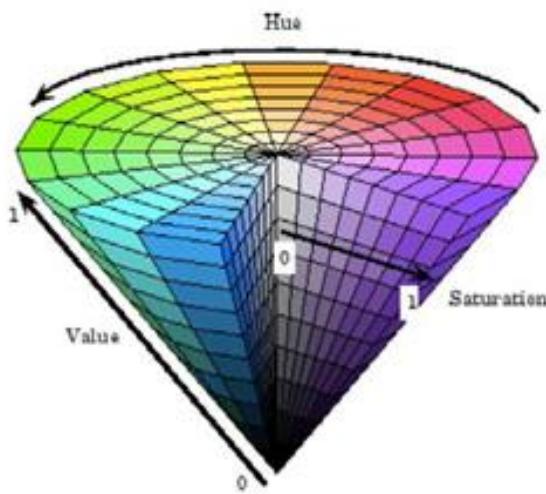
באלגוריתם זה השתמשנו במספר טכניקות של עיבוד תמונה:

- HSV – אלגוריתם לזיהוי צבע.
- פעולות מורפולוגיות.
- מציאת הקונטור של הcadור.

הקלט של האלגוריתם הוא פריים של הסרטון שמיוצג בRGB.



תחליה אנו ממיררים את התמונה מייצוג RGB ליצוג HSV:  
**HSV**



מרחב הצבעים HSV (נקרא לפעמים גם LHS) במודל זה כל צבע מיוצג על ידי גוון(Hue), רווייה(Saturation), ערך(Value).

זו בעצם השיטה הכי אינטואיטיבית לייצוג צבע.

- **גוון**- מייצג את הגוון של הצבע עצמו, הגוון מיוצג לעיתים קרובות כזווית בגלגול צבעים(הגוון האדום הוא בעצם ערך 0 , צהוב זה בערך 60, ירוק זה 120 , כחול זה 240, ואדום זה בחזרה 360).
- **רוייה**- מספר אשר מיוצג בין 0 ל 1 (בפועל דומה לערבוב צבע הבסיס עם גוש לבן) כאשר 0 מייצג צבע לבן ו 1 מייצג צבע שלא אורבב בכלל עם לבן.
- **ערך**- מספר בין 0 ל 1 אומר בעצם עד כמה הצבע מואר או חשוך.

לאחר מכן אנו מבצעים 2 מסיכות שיסנוו את הצבע האדום מהתמונה.

השתמשנו ב 2 מסיכות מהסיבות הבאות:

אובייקטים אדומים עשויים להופיע לעיתים בגוונים שונים עקב תנאי תאורה שונים. שימוש בשתי מסכות מבטיח זיהוי מיטבי בסיטואציות שונות, ובכך האלגוריתם הופך גמיש ויעיל יותר ומגדיל את האמינות של משימת זיהוי הצבע האדום.

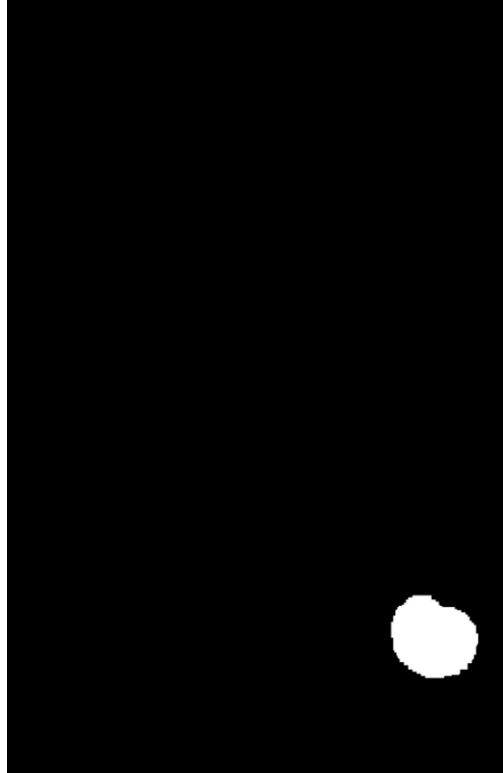
לאחר פועלה זו התקבלה תמונה בינהית שמסמנת את האזוריים האדומים בתמונה.



לאחר מכן אנו משתמשים במסכות מורפולוגיות על מנת לנוקות רעשים ולהחליק את הcador.

### פירוט מלא על הפעולות המורפולוגיות יינתן בהמשך בחלק של אלגוריתם זיהוי הרצפה

ו התקבלה התמונה הבאה אשר אמורה לייצג רק את הcador:



#### מציאת קונטורים בתמונה:

קונטורים (קווי מתאר) - הם קוים רציפים או עקומות אשר מתארים את היקף של האובייקטים בתמונה.

הfonקציה cv2.findContours משמשת לזיהוי קוים אלה בתמונות ثنאיות, כאשר הפיקסלים הם 0 או 255. פונקציה זו שימושית במיוחד בזיהוי אובייקטים, ניתוח צורות וחילוץ תכונות.

## תיאור דרך פעולה של cv2.findContours:

- האלגוריתם מקבל תמונה בינהרית.
- התמונה נסרקת מהצד השמאלי העליון לצד ימין תחתון של התמונה, פיקסל אחר פיקסל עד שmaguius לפיקסל לבן אשר מסמן הימצאותו של אובייקט.
- האלגוריתם עוקב אחר גבול האובייקט על ידי הסתכילות על 8 השכנים המוחברים של הפיקסל (פיקסלים צמודים ישירות אופקית, אנכית או אלכסונית) וקובעת איזה שכן הוא חלק מקו המתאר.
- מאחסן כל קו מתאר שזוהה בתמונה כרשימה של נקודות.

ניתן לאחסן את כל קווי המתאר שזוהו בזיכרון, אך אחסון כל פיקסל בקו המתאר עשוי להיות עתיק זיכרון דבר אשר יכול להאט את האלגוריתם.

1. מכיוון שהמשחק שלנו הוא אינטראקטיבי, ריצת האלגוריתם בזמן אמת היא נשא קרייטי, לכן אנו בחרנו בשיטת SIMPLE\_CHAIN\_APPROX cv2.CHAIN\_APPROX\_SIMPLE אשר חוסכת בזיכרון על ידי הסרת כל הנקודות המיותרות ומחיסת קווי המתאר, ובכך מאחסנת רק נקודות קרייטיות ליהיו קו המתאר.
2. כיוון שאנו רוצים לזהות את היקף הcador בלבד, בחרנו בשיטת RETR\_EXTERNAL – שיטה זו מחזירה רק את קווי המתאר החיצוניים ביותר. זה שימושי כאשר מעוניינים רק בצורה החיצונית של האובייקטים, תוך ה
תעלמות מכל קווי מתאר שיכולים להימצא בתוך אותם אובייקטים. מצב זה יכול להפחית משמעותית את מספר קווי המתאר שנמצאו והוא מהיר יותר ביחס לשאר.

לאחר שקיבלנו את כל הקונטורים אשר מופיעים בתמונה, אנו בוחרים בקונטור הגדל ביותר, כיוון שזה הקונטור שמייצג את הcador(בנהנעה שאין אובייקטים גדולים נוספים בתמונה).

השלב הבא הוא מציאת המעל הקטן ביותר אשר מכיל את כל קו המתאר המקסימלי (שזוהה קודם לכן) מכאן אנו מחלצים באופן חד חד ערכי את נקודות מרכז המעל ואת רדיוסו של הcador.

נשים לב שבחרנו לסמן רק את הקונטור המקסימלי שרדיוiso גדול מערך סוף מסוים אשר מצאנו על ידי כך שהרחקנו את הcador למרחק המקסימלי מהמצלמה שנitin בגבולות המשחק ומצאנו את רדיוסו במקרה זה.

סיבה נוספת לשימוש בתנאי היא על מנת לסנן מקרים בהם גוון עורו של השחקן נתפס כאדם מסוים וכך שעלול להיחשב כcador.

כיוון שביחס לכדור כל עיגול על גופ השחקן הוא קטן וכן האלגוריתם יתעלם מהעיגולים שיימצאו על גופ השחקן.

לבסוף סימנו בכל פריים את המעלג שמצאנו אשר מייצג את מיקום הcadour בכל רגע נתון.  
יש לציין שיצרנו מערך positions שכל איבר במערך מכיל את נקודת מרכז המעלג ובכל פריים דגם הוספנו איבר חדש למערך. מערך זה ישמש אותנו למימוש אלגוריתם לזריהו הקפזה קקלט האלגוריתם.



**אלגוריתם לזיהוי הקפצת כדור:**

**רקע ומטריבציה:**

חלק עיקרי במשחק הקפיצות הוא לאזהות הקפצת כדור תקינה, על מנת שנוכל לספר את כמות הקפיצות ולהציג לאורך המשחק ובסופו.

**גלוֹט האלגוריתם** – שלוש איברים סמוכים של  $dx_i$   $dy_i$  שמתאר את מרכז הכדור בכל פריים.

**פלט האלגוריתם** – זיהוי הקפצה ועדכון כמות הרקציות הנוכחיות.

**מהלך האלגוריתם:**

תחילת חישבונו את המרחק שעבר מרכז הכדור בין כל שני פריים סמוכים

$$dx_1 = x_1 - x_0, dx_2 = x_2 - x_1$$

$$dy_1 = y_1 - y_0, dy_2 = y_2 - y_1$$

נשים לב שהמרחק שמעניין אותנו כרגע הוא המרחק בציר  $y$  שכן אם  $i = 1, 2$ ,  $dy_i$  חיובי איזה כדור בירידה, ולהיפך אם  $i = 1, 2$ ,  $dy_i$  שלילי הכדור נמצא בעלייה. הקפזה תהיה מצב שבו  $dy_1$  יהיה חיובי ו  $dy_2$  יהיה שלילי, لكن התנאי של זיהוי גרדיאנט מספיק בכיוון הנכון יהיה:

- $-3 < dy_2 < 3$
- $dy_1 > 2$
- $dy_1 \cdot dy_2 < 0$

נשים לב שתאת הערכים 2 ו-3 – בחרנו לאחר ניסוי וטעיה של גרדיאנט מספיק שלא יהיה זיהוי שווה של הקפזה עקב תנוצה קטנה מדי.

על מנת לאזהות הקפזה נרצה רק יהיה שינוי בגרדיאנט אלא גם שינוי חד ב מהירות של הכדור כזכור שה汰וצה שלו תהיה גבוהה מספיק לנו ליצור  $speeds$  שכלי איבר בו הוא מהירות הכדור בכיוון  $x$  ובכיוון  $y$ . נשים לב שהזמן שעובר בין כל זוג פריים הוא 1.

ולכן  $v_{y_i} = dy_i$  ו  $v_{x_i} = dx_i$  לכל  $i$ .

על מנת לחשב את התאוצה בכיוון  $x$  ובכיוון  $y$  לכל זוג פרימיטים נחסיר את המהירות בין כל זוג פרימיטים בהתאם . $v_{x_i} - v_{x_{i+1}} = a_{x_i}$  |  $v_{y_i} - v_{y_{i+1}} = a_{y_i}$

על מנת לחשב את התאוצה השקולה נבצע:

$$\hat{a} = \sqrt{a_{y_i}^2 + a_{x_i}^2}$$

את התאוצה השקולה שיחסבנו נשווה לערך סף ובמידה זהה גדול מערך הסף נסיק שהתקיים התנאי השני להקפזה שאומר שבזמן הקפזה הcador מוקפץ על ידי רגלו של השחקן כלומר תאוצתו גדולה.

לכן התנאי של זיהוי שינוי מהירות מסויך יהיה:

$$\bullet \quad \hat{a} > 10.$$

נשים לב שאת הערך 10 בחרנו לאחר ניסוי וטעיה של שינוי המהירות.

כאשר שני התנאים מתקיים נסיק שקרתת הקפזה.

.if gradient\_bounce\_detected and velocity\_bounce\_detected

### **זיהוי פגיעה ברצפה:**

### **רקע ומוטיבציה:**

אחת המטרות שצינה לעלה הייתה לזהות את הרצפה ולהפריד אותה מן השחקן, האתגר היה שכאשר הcador קופץ על הרצפה, הקפיצה נספרת כהקפיצה עקב אלסטיות cador הcadorgal.

**קלט האלגוריתם** - שני פרייםים עוקבים מצלמת הוידאו.

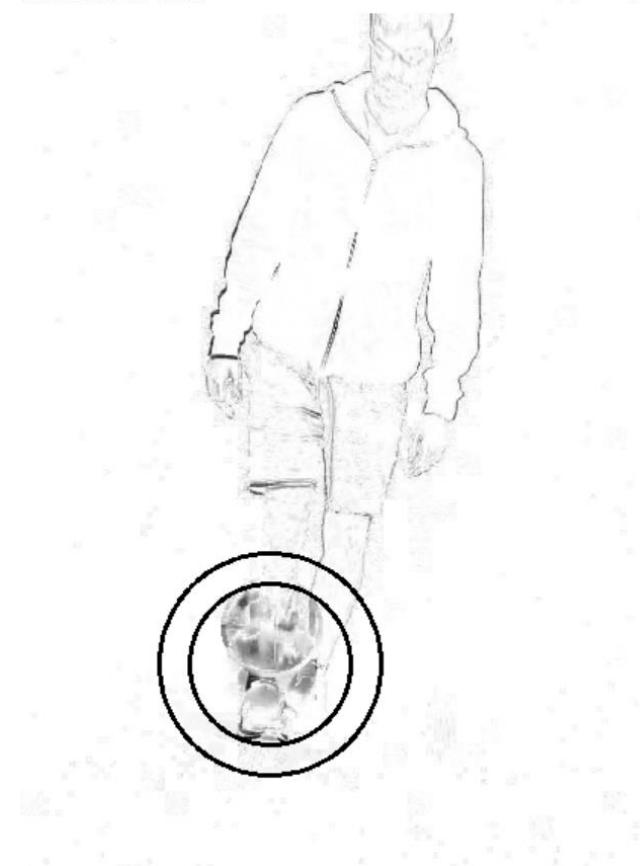
**פלט האלגוריתם** – תמונה הפרשים שעלייה מסומנת טבעת סביבcador.

### **מהלך האלגוריתם:**

ראשית, זיהינו את הרצפה ע"י שימוש בעובדה שהרצפה סטטית לאורך כל המשחק, ואילו השחקן והcador בתנועה.

על מנת לבצע מטרה זו, בחרנו להשתמש בתמונות הפרשים כך שכאשר שני פיקסלים בעלי מיקום זהה בין שני פרייםים עוקבים, יהיו בעלי ערך זהה והיחס בין שני הפיקסלים יהיה אפס, ואילו מפני שהשחקן והcador Zusim ערכיהם הפיקסלים בין שני פרייםים עוקבים יהיו שונים ולכן היחס יהיה שונה מzero.

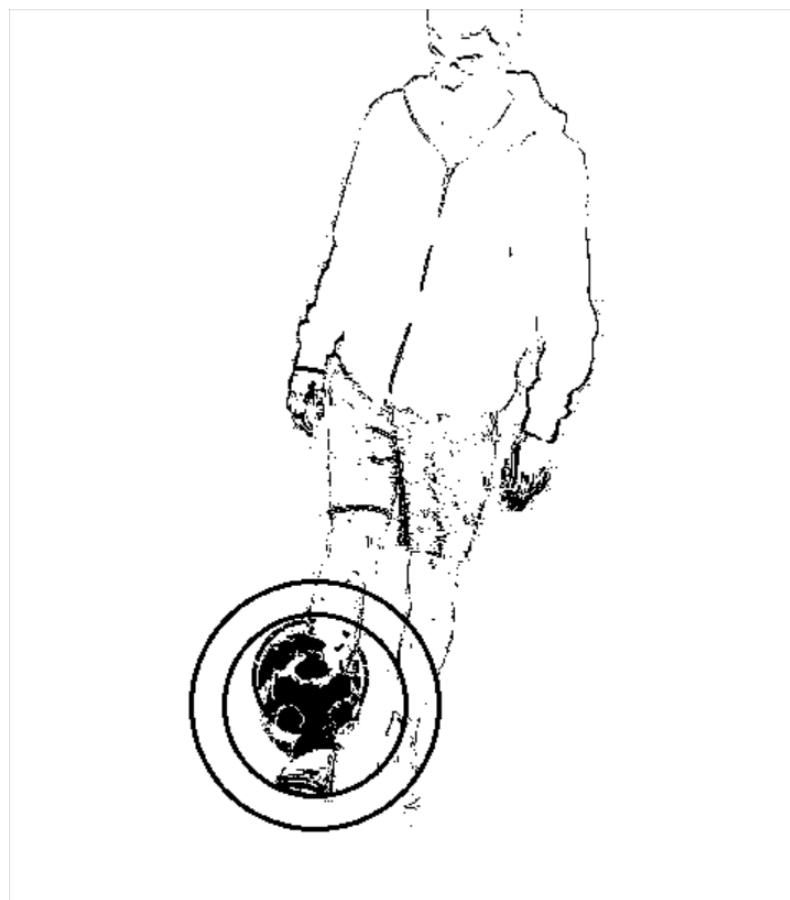
לאחר תמונה הפרשים, קיבל אמן הפרדה מסוימת של הרצפה מן השחקן- אולם קיבל תמונה רועשת מאוד שלא אפשר לנו להחליט האםcador פגע ברצפה.



בתמונה ניתן לראות כי אכן החלקים הסטטיים בתמונה הם בגוונים לבנים יותר, ואילו החלקים שזרים בגוונים שחורים יותר, זאת כיוון שהפעלו אופרטור **tot** על תמונה ההפרשים לצורכי אסתטיקה ונוחות הצגה.

יחד עם זאת, קיבלנו תמונה לא בינהרית שמכילה ערכים אפורים, מה שמקשה על ביצוע האלגוריתם, ולכן דבר ראשון שעשינו היה להפוך את התמונה לבינהרית באמצעות קביעת threshold מתאים כך שערכי פיקסלים שמתחת הסף יהפכו לבנים באופן מוחלט וערכי פיקסלים שמעל הסף יהפכו לשחורים באופן מוחלט.

לאחר קביעת ערך הסף קיבלנו תמונה זו:



כעת אכן קיבלנו תמונה בינהרית, אך נשים לב שללא שימוש בשום פילטר נקלט רגשות גבואה לשינויים בין פרימיטים, כך שאפילו שינוי קלים שלא דוקא נובעים מהתנועה עצמה אלא מגורמים כמו אור וצל. ישפיעו על תמונה ההפרשים.

בנוסף, קיבלנו תמונה עם הרבה חורים, מה שמקשה על הפרדה מיטבית בין השחקן והבדור לרצפה.

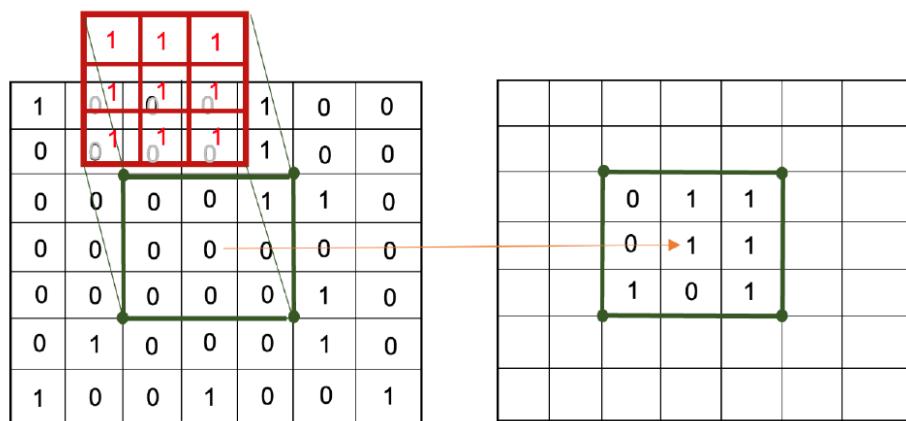
כדי להתמודד עם תוצאות אלו, הפעלנו על התמונה הבינארית פעולות מורפולוגיות של dilate, erosion ושלובם - close.

לצורך כך, תחילה הגדרנו חלון בגודל  $3 \times 3$  של אחדות ובאמצעותו הפעלנו את הסיגנון.

זהה פועלות הרחבה.

הרחבה מתבצעת על ידי קונבולוציה של התמונה עם גרעין הינו box-filter סטנדרטי.

אם לפחות פיקסל אחד מתחת לגרעין הינו שחור, אז הפיקסל בתמונה יהיה שחור גם כן, כמווארה בacz הכתום בתרשים.



את פועלות הרחבה ניתן לעשות מספר פעמים וכך לקבל הרחבה משמעותית יותר כאשר מספר האיטרציות גדול יותר.

הפעלנו את פועלה זו למשך 3 איטרציות וקיבliśmy את תוצאה הבינאים הבאה:



נשים לב שבתמונה לעיל קשה לזהות את השחקן, בפרט את רגלי השחקן וגם את הcador- שנראים כמקשה אחת.

יחד עם זאת, הצלחנו להעלם בצורה טובה (אך עדין לא מושלמת) את רשי הרקע שמסביב.

על מנת להתמודד עם בעיה זו, נוסיף גם את פועלות ה-*erosion*.

כשחזר רק אם כל הפיקסלים בגראון שחורים גם כן. **Erosion**- זהה פועלות מצומצם. פועלה זו הפוכה לפועלות *dilation*, כך שנסמן פיקסל

**Close**- היא פעולה שמלבת את פועלות הרחבה (*dilate*) ופעולות הצמצום (*erosion*), כלומר, תחיליה נרחבת האזוריים השחורים בתמונה ולאחר מכן נחזר אותם לגודלם המקורי. פועלות *closing* מסייעת גם במילוי חורים.

לאחר פעולה זו נקבל את התוצאה הסופית והרצויה:



כפי שניתן לראות, לאחר כל הפעולות שהפעלנו, קיבלנו תמונה הפרושים אינפורטטיבית, שמספרידה לנו את שתי הרגליים. בתמונה זו אין הפרדה בין הcador לרגל, אבל היא צולמה במהלך הקפיצה ולכן זה צפוי ורצוי.

## הגדרת הטבעת ושימושה לצורכי זיהוי הרצפה-

לאחר שקיבלנו תמונה הפרשנים מספיק טובہ שמספרידה בין האזוריים הסטטיים בתמונה לבין האזוריים הדינמיים ללא השפעת אור וצל ותנועות חדשות, נרצה לישם באמצעותה את זיהוי נפילת הcador על הרצפה.

לצורך כך, הגדרנו טבעת שרדיוישה הפנימי הוא רדיוס הcador+מרחק ביטחון, ורדיוישה החיצוני גדול ב-20 יחידות מהרדיוישה הפנימי.  
את המידע על מיקום הcador אנו מקבל מהאלגוריתם הראשון שמצוה את הcador ונוטן מידע על מרכז הcador ורדיויסו בכל פריים ומקייף אותו בעיגול.  
הסיבה שבחרנו לחתך מרחק ביטחון עבור הרדיוס הפנימי של הטבעת היא שהקפת הcador זהה יחד עם הcador בין הפריים ובתמונה הפרשנים התוצאה היא טבעת שחורה לא רצiosa שפוגעת בבדיקה האלגוריתם לזיהוי הפגיעה ברצפה.

לאחר שהגדרנו את הטבעת, נרצה לחשב את אחוז הפיקסלים השחורים בטבעת. הסיבה לבחירה בשיטה זו היא על מנת להפריד בין הקפצה אמיתית של השחקן להקפצה על הרצפה, כאשר אם השחקן שנע מקפץ את הcador אחוז הפיקסלים השחורים בטבעת יהיה יותר גדול בצורה משמעותית מהמצב שבו הcador קופץ על הרצפה, מכיוון שבעת הקפצה, הרgel הנעה של השחקן תיכנס לטבעת.  
(זכרו - פיקסלים שחורים מייצגים את החלקים הנעים בתמונה ואילו פיקסלים לבנים מייצגים את החלקים הסטטיים).



פילטר הטבעת אשר מקיפה את הcador  
לאורך המשחק

חישוב אחוז הפיקסלים השחורים בטבעת הטעבע באמצעות סכימת הפיקסלים השחורים בטבעת (פעולות `and` `bitwise` בין הפיקסלים השחורים של תמונה הפרשנים לבין הפיקסלים הלבנים במסכת הטבעת - כלומר הטבעת עצמה), סכימת כל הפיקסלים בטבעת, וחילוקה בין הסכימות הללו.  
הינו צריכים להגדיר סף שמתיחסו לטבעת תיחשב "לבנה" ולאחר אנליה ראיינו שערך הסף המתאים הוא 25 אחוז.

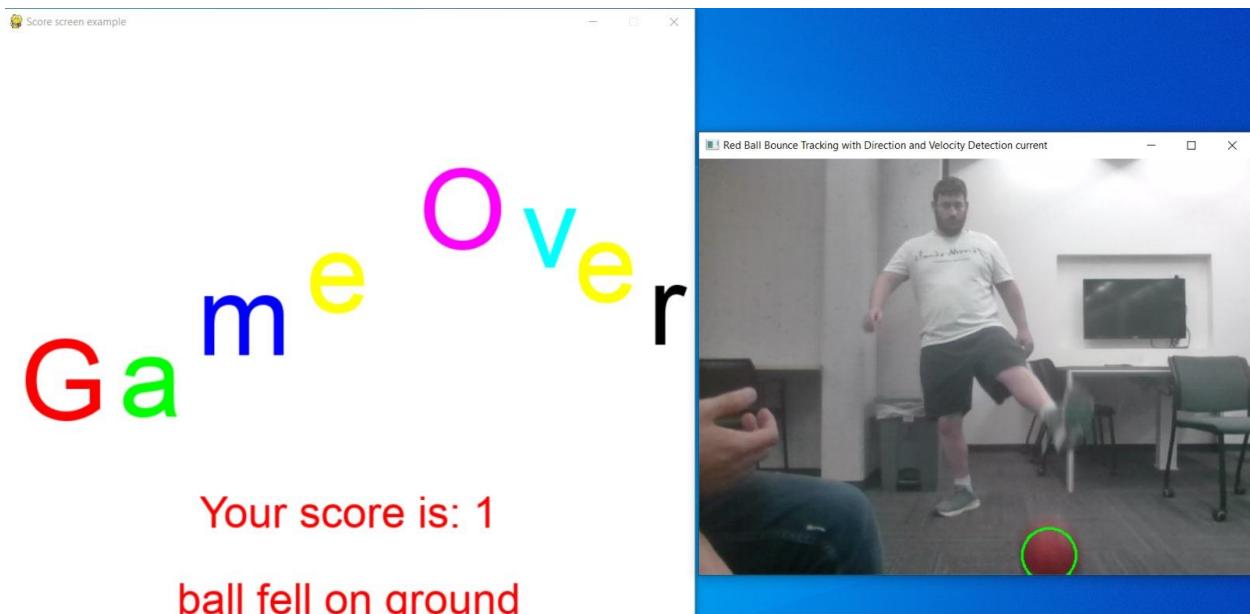
אולם, לצורך קביעה פסילה לא נוכל להשתמש רק במידת הטעות, מכיוון שפעמים רבות כאשר ה cedar נמצא בתנועתו באוויר והruk מאחריו סטטי (לדוגמה כאשר השחקן משמאלי או מימין ל cedar).

לכן נדרש שיתקיים שלושה תנאים:

- התרחשה הקפיצה, שכן cedar תמיד קופץ לאחר פגיעתו ברצפה.
- הטעות תיחס כלבנה (אחוז הפיקסלים השחורים יהיה מתחת ל-5%).
- cedar ימצא בחצי התחתון של התמונה.

רק אם כל שלושת התנאים הללו יתקיימו, נזהה את הקפיצה כפגיעה ברצפה ונפסול את השחקן.

לבסוף כשזהותה נפילה על הרצפה עולה הודעת הפסילהoSיבתה, כמו כן מוצג גם מספר הקפיצות שנצברו עד כה. בנוסף לכך מוצגים פרטיים הפסילה.



## **אלגוריתם זיהוי נגיעה ביד רקע ומוטיבציה:**

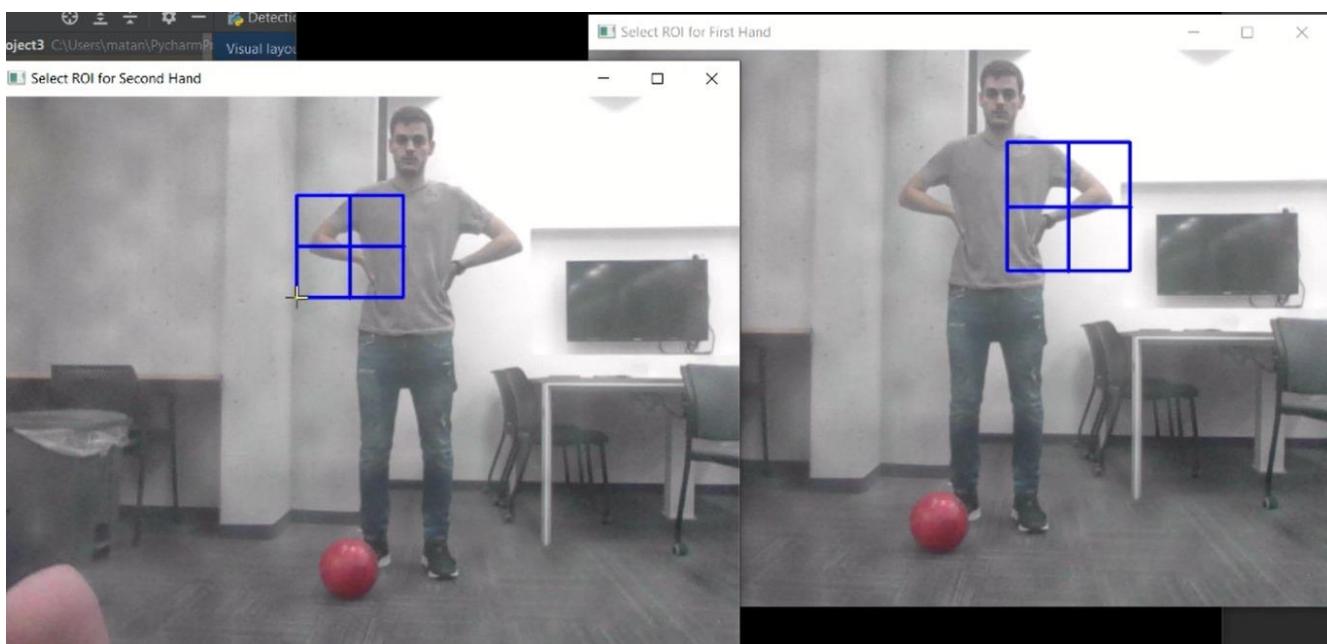
במשחק הcadorgal נגעה של היד בצד ימין חיונית שכן קבענו כי נגעת היד בצד ימין במהלך המשחק תוביל לפסילה. שכן שימוש ביד במהלך המשחק ההפצות פוגם במטרת המשחק שהוא שיפור יכולות השחקן וספרה של ההפצות.

**קלט האלגוריתם** – זיהוי הקפזה חיובית, שלוש איברים סמוכים של מערכות positions שמTARGET את מרכז הcador בכל פריטים, מיקומי ריבועי הידיים שסומנו בתחילת התהילה ומटעדכנים על ידי CSRT.

**פלט האלגוריתם** – הודעת פסילה לאור נגעה ביד הספציפית (שמאל או ימין) וכמות ההפצות החזיות שניצבו עד כה.

### **מהלך האלגוריתם:**

בתחילת המשחק מנהל המשחק מסמן את ידיו של השחקן בשתי מלבנים באופן ימני, כאשר יד שמאל מסומנת ראשונה ויד ימין מסומנת לאחריה. האלגוריתם מקבל את שMONOTONIE קודקודיו המלבנים בהתאם. לאחר מכן מתבצע מעקב אחרי ידי השחקן על ידי אלגוריתם CSRT.



## : CSRT

אלגוריתם המקבט CSRT ( Discriminative Correlation Filter with Channel and Spatial Reliability ) הוא אלגוריתם מתקדם למקבט אחר אובייקטים המשמש בישומים של ראייה ממוחשבת כדי לעקוב אחר תנועת אובייקטים ספציפיים בתוך הווידאו.

### מושגים מרכזיים בCSRT

מסנני קורלציה : אלגוריתם CSRT משתמש במסנני קורלציה כדי לקבוע את מיקומו של אובייקט בתמונה. **Tracker** משתמש במסננים אלו כדי ליצור מודל של האובייקט על סמך המראה שלו, אשר מתעדכן במהלך הזמן.

מבחן מתמטית נגיד  $(t)x$  המיצג את הפיקסלים המשויכים לאובייקט שנמצא במלבן הראשוני המסומן בזמן  $t$ .

נסמן את פילטר הקורלציה כ- $(t)h$  והוא מחושב על ידי ביצוע התמרת פורייה של  $(t)x$  והכפלה בצדד שלו.

$$H(w) = \mathcal{F}[x(t)]\overline{\mathcal{F}[x(t)]}$$

ולאחר ביצוע התمرة הפוכה ניתן לקבל:

$$h(t) = \mathcal{F}^{-1}[H(w)]$$

על מנת למצוא את האזור בפריים החדש שתואם בצורה הטובה ביותר ביותר לאובייקט אנו מפעילים את פילטר הקורלוציה  $(t)H$  על הפריים החדש  $(t)y$ .

בצורה הבאה:

$$C(t) = \mathcal{F}^{-1}[H(w)\mathcal{F}[y(t)]]$$

מקבט בהתבסס על ערוצים: אלגוריתם CSRT משתמש במספר ערוצי תמונה (כגון צבע, גרדיאנטים) כדי לעקוב אחר האובייקט. כל ערוץ מספק פרספקטיבית ייחודית על תכונות האובייקט (פיצ'רים), והשימוש במספר ערוצים משפר את יכולת העקיבה אחר האובייקט הרצוי, ניתן לעקוב אחר אובייקטים שונים במצבים שונים ומגוונים לבחירתנו.

אמינות בערזץ: לא כל הערוצים תורמים באותה מידת למקבט יעיל. CSRT מעריך את האמינות של כל ערוץ ונותן לכל ערוץ ציון. הערוצים עם האמינות הגבוהה כלומר בעלי ציון גבוה יותר ישפיעו יותר על תהליכי המקבט .

מבחן מתמטית נגדיר את  $(t)_i c$  כתגובה הקורולציה עבור הערוץ  $i$  בזמן  $t$ . ניקוד האמינות  $(t)_i s$  לכל ערוץ מחושב על ידי תגובה הקורולציה

$$s_i(t) = \frac{1}{N} \sum_{j=1}^N c_i(t, j)$$

כאשר  $N$  זהו מספר הפיקסלים בתחום המלבן התוחם.

האמינות הערוץ  $(t)_i r$  עבור ערוץ  $i$  נקבעת על בסיס הניקוד  $(t)_i s$  ועל גבי ניקוד סך הערוצים:

$$r_i(t) = \frac{s_i(t)}{\sum_{j=1}^M s_i(t, j)}$$

כאשר  $M$  זהו מספר הערוצים הכללי.

אמינות מרחבית: אלגוריתם CSRT מתייחס גם לאזוריים מרחביים של האובייקט. חלקים מסוימים יכולים להיות יותר אמינים למשך מאחרים בגליל יציבות הרקע, ייחודיות האזור או גורמים אחרים. רכיב זה מסייע לשימור על פוקוס במשך אחר חלק האובייקט המרכזיים, גם אם אזוריים אחרים מושתטים או מוטשטשים.

מבחן מתמטית נגדיר את  $(t)_j P$  אשר מגידר את ניקוד האזור המרחבית בערוץ  $j$  בזמן  $t$ . נגדיר את  $(t)_Q$  להיות מדד האמינות המרחבית הכלול והוא נקבע על ידי ניקוד האזוריים המרחביים השונים.

$$Q(t) = \sum_{i=1}^M \sum_{j=1}^K P_{ij}(t)$$

כאשר  $K$  מתאר את מספר האזוריים המרחביים על גבי כל הערוצים.

## כיצד CSRT עובד:

אתחול: בתחילת המпроיקט, CSRT מקבל אזכור אותו אנו מגדירים כמלבן מסביב לאובייקט למשך. מלבן זה מסומן על גבי הפריים הראשון של הסרטון.

יצירת מסנן הקורלציה: CSRT יוצר מסנן קורלציה על סמך האובייקט בתוך המלבן המסומן. מסנן זה משמש לזיהוי ומעקב אחר האובייקט בתמונות הבאות.

### מעקב בין פרייםים:

CSRT מחליט את מיקום האובייקט הטוב ביותר ביותר בפריים החדש באמצעות מסנן הקורלציה. הוא משתמש במספר ערכזים תמונה כדי לשפר את הדיק, ומתחשב בשינויים במצב, בטקסטורה או בגרדיינט.

ה**Tracker** מציין ציון אמינות לכל ערוץ ולכל אזור מרחבית בתוך האובייקט. זה עוזר ל-CSRT להתאים את המיקוד על החלקים הći אמינים.

עדכן המודול: ככל שהאובייקט נע והרקבו של האובייקט שבתוכו הריבוע משתנה, CSRT מעדכן את מודל המסנן קורלציה שלו כך שיישקף את השינויים הללו. זה מבטיח **Tracker** יתאים לשינויי מיקום האובייקט.

פלט של אזכור המпроיקט המתעדכן: לאחר העיבוד של התמונה, CSRT מחשיר מלבן עדכני שמציאן את המיקום המעודכן של האובייקט. מלבן זה ישמש להמשך המпроיקט בתמונות הבאות.

### יתרונותיו של אלגוריתם ה-CSRT:

גמישות: השילוב של שימוש בערכזים אמינים שונים ובאמינות באזוריים מרחביים שונים מספק יכולת התמודדות עם מצבים קיצוניים שונים כגון: שינוי תאורה, טשטוש וראות חלקית של האובייקט (קשהים שאיתם התמודדנו בפרויקט).

הסתגלות: ה-CSRT מתאים את עצמו למראה המשנה של האובייקט לאורך הזמן, מאפשר מпроיקט רציף גם כאשר התנאים משתנים.

לסיכום, הוא **Tracker** גמיש וחזק המתאים למגוון תרחישים של מпроיקט אחר אובייקט.

### **הتلבותות לשימוש באלגוריתם המקב**

מצין כי התלבטו בין שני אלגוריתמים למשך תנועות הידיים בין הפרויימרים בזידאו והם : CSRT tracker ו optical flow (Lucas-Kanade) .

האלגוריתם שאיתו בחרנו למש את המשימה הוא CSRT tracker מכמה סיבות:

- העקב עקב שימוש תנועת האובייקט הנעקב בעריצים שונים ובאמינות הרחבה.
  - עמידות לפני רקעים משתנים ומורכבים.
  - עקיבת אינטואיטיבית גם כאשר ישנה חסימה חלקית של האובייקט הנעקב.

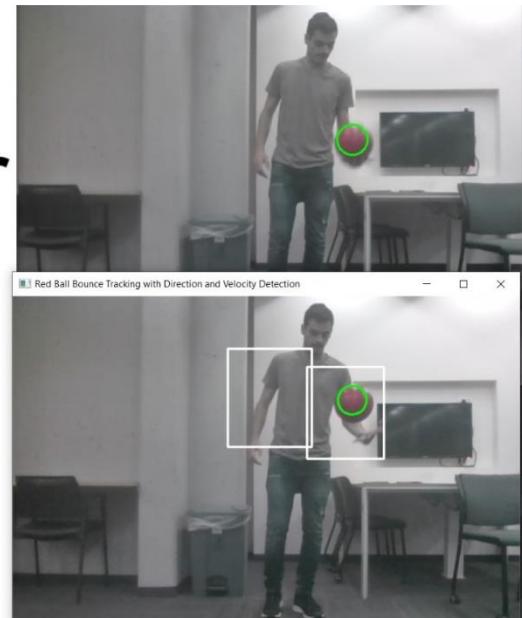
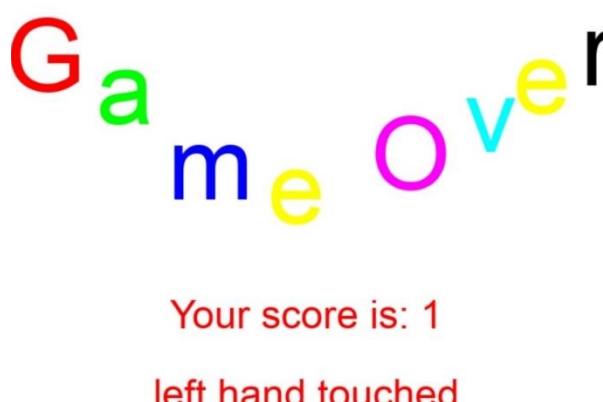
## זיהוי פסילה עקב נגיעה ביד

באופן סימולטני למקבב הידיים, האלגוריתם משתמש יחד עם האלגוריתם לזריהו ההפצה. כאשר מתקיים התנאי על קיום הקפיצה נכנסים אל האלגוריתם לבדיקה של חוקיותה.

תחילה נבדוק האם קיימים בהקפה הנוכחית זיהוי של הcador, לאחר מכן נבדק האם מיקומי מרכז הcador הם בהתאם למבצע היד השמאלית.

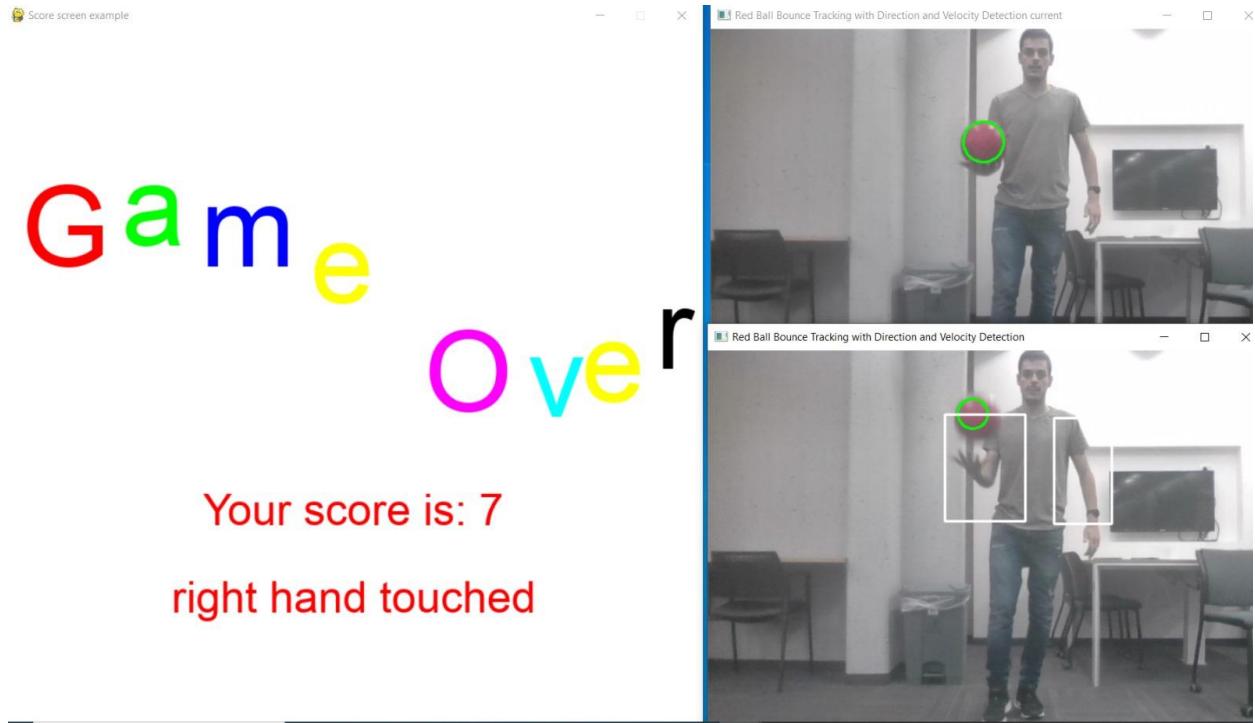
$((p1[0] \leq x \leq p2[0]) \text{ and } (p1[1] \leq y \leq p2[1]))$

ברגע שזוהההה הקפצה בתוך מלבן היד אנו מסיקים כי התבצעה נגעה ביד השמאלית. פושלים את השחקן ומציגים לו את מספר הרקפות שלו.



אותה בדיקה מתרבצת עברו היד اليمنית לאחר מכן.

$$((p3[0] \leq x \leq p4[0]) \text{ and } (p3[1] \leq y \leq p4[1]))$$



#### תצוגת הטבלה הסופית:

לבסוף נציג את טבלת המקפיצים כך שנראה את שם ושם משפחתו של המקפץ ואת מספר ההקפות אותן השיג. בסיום המאורע(יום הפראייקטים) נזכיר על המנצח הסופי.

	First Name	Last Name	Score
1	matan gab	gabai	56

## אנליזת התוצאות - בוחנת ביצועי אלגוריתם זיהוי הרצפה

**הקדמה לניטויים:** כדי לאמוד את ביצועי אלגוריתם זיהוי הרצפה שלנו, בחנו מספר קטעי וידאו של מהלך המשחק ב"תנאים קיצוניים", ובחנו את השפעת התנאים הקיצוניים על זיהוי הרצפה.

התנאים הקיצוניים כוללים:

- A. נפילת הcador במרחק קטן מאוד מרגלי השחקן כר שבפריים נראה כאילו השחקן הקפיץ את הcador אך בפועל הcador נפל על הרצפה, ביחס למצב בו הcador נפל במרחק מסויך גדול מרגלי השחקן כר שהאלגוריתם לא לוקח בחשבון את השחקן.
  - B. נפילת הcador על הרצפה מלפני השחקן כר שבפריים השחקן משפייע על האלגוריתם (השחקן גורם לאלגוריתם הփרסים עקב תזוזתו בתוך הטבעת אותה הגדרנו קודם בפירוט האלגוריתם) ביחס למצב בו הcador נפל לצד השחקן כר שرك כף רגלו של השחקן עלולה להשפייע על האלגוריתם בכך שכף רגלו של השחקן נכנסה לטבעת המוגדרת. תנאים אלה עשויים להשפייע על ביצועי האלגוריתם ועל יכולת שלנו לזהות בזורה מהימנה את הנפילה של הcador על הרצפה.
- על מנת לבדוק את איקות ה- $p$ -threshold שקבענו ערכנו ארבעה ניטויים, שכל ניטוי מייצג את אחד מקרים הקצה הנ"ל.

**ניסוי ראשון:** בדקנו את נכונות סיווג האלגוריתם עבור המקרה בו הcador נפל קרוב לרגלי השחקן ומ לפניו.

**ניסוי שני:** בדקנו את נכונות סיווג האלגוריתם עבור המקרה בו הcador נפל קרוב מרגלי השחקן ולצדו.

**ניסוי שלישי:** בדקנו את נכונות סיווג האלגוריתם עבור המקרה בו הcador נפל רחוק מרגלי השחקן ומ לפניו.

**ניסוי רביעי:** בדקנו את נכונות סיווג האלגוריתם עבור המקרה בו הcador נפל רחוק מרגלי השחקן ולצדו.

כל ניטוי ביצענו שלוש פעמים עבור  $thresholds$  פוטנציאליים שונים (15, 25, 35), על מנת לבחור את ה- $p$ -threshold האידיאלי.

**השערת הניסויים:** לפי הצורה בה מימושנו את אלגוריתם זיהוי הרצפה, ההערכה שלנו היא שהביצועים עברו נפילת הבדיקה במרקחן קטן מרגלי השחקן או בדיקת מלפנים לשחקן היו פחות טובים מאשר הביצועים עברו נפילת הבדיקה במרקחן רב מרגלי השחקן או לצד השחקן, כלומר אחוז הפיקסלים השחורים בטבעת (המייצגים את האזוריים הקיימים בפריים) יהיו גדולים יותר. זאת כמובן מכיוון שזיהוי נפילה על הרצפה מתבסס על תמונה ההפrixים ועל ידי הפרדה בין חלקים ניחים בתמונה לחלקיםZZים בתמונה. لكن ככל שיהיה יותר "מרוח ביטחון" של הבדיקה מחלקיםZZים בתמונה, נזהה בצורה מדויקת יותר את נפילת הבדיקה על הרצפה.

השערה נוספת שלנו הייתה שהיא threshold האידיאלי מבין שלושת האפשרויות יהיה האמצעי מבניהם(25).

**דרך פעולה:** עברו כל אחד מהניסויים השתמש ב-xmatrix confusion matrix, כפי שראינו בהרצאות הקורס.

עבור כל ניסוי, יצרנו מאגר של סרטונים הרלוונטי לאותו ניסוי, ובדקנו את אחוז הדיקון בהתאם.

נשים לב שעבור נפילה אמיתית על הרצפה ישן שתי אפשרויות:

- TP- זיהוי נכון של ההקפצה כפגיעה ברצפה
- FN- זיהוי כזב של ההקפצה כהkpצה חוקית, למרות שבפועל הייתה נפילה על הרצפה.

באופן דומה, עבור הקפצה חוקית ישן שתי אפשרויות:

- TN- זיהוי נכון של ההקפצה כהkpצה חוקית.
- FP- זיהוי כזב של ההקפצה כפגיעה ברצפה, למרות שבפועל הייתה הקפצה חוקית.

עבור כל אפשרות נזאת מדדנו את המספר היחסי שהמאורעות התרחשו, ביחס למספר הניסיונות, או במילימטרות- חישבנו את אחוז השגיאות עבור כל שגיאה (FN,FP) ואת אחוז ההצלחות עבור כל הצלחה (TP,TN).

$$ACC = \frac{TP+TN}{TP+TN+FP+FN}$$

### תוצאות הניסוי:

עבור כל ניסוי יצרנו 25 סרטונים בהם הcador נפל על הרצפה ו-25 סרטונים בהם הcador הוקפץ על ידי השחקן.

#### **ניסוי ראשון – נפילתcador קרוב לרגלי השחקן ומפניו**

עבור threshold=15

FN	FP	TN	TP
11	0	25	14

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} = \frac{14 + 25}{50} = 78\%$$

עבור threshold=25

FN	FP	TN	TP
8	1	24	17

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} = \frac{17 + 24}{50} = 82\%$$

עבור threshold=35

FN	FP	TN	TP
5	5	20	20

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} = \frac{20 + 20}{50} = 80\%$$

## ניסוי שני – נפילת הcador קרוב לרגלי השחקן ולצדיו

עבור threshold=15

FN	FP	TN	TP
6	1	24	19

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} = \frac{19 + 24}{50} = 86\%$$

עבור threshold=25

FN	FP	TN	TP
4	2	23	21

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} = \frac{21 + 23}{50} = 88\%$$

עבור threshold=35

FN	FP	TN	TP
4	3	22	21

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} = \frac{21 + 22}{50} = 86\%$$

### **ניסוי שלישי – נפילת הcador רחוק מרגלי השחקן ומולפנוי**

עבור threshold=15:

FN	FP	TN	TP
5	1	24	20

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} = \frac{20 + 24}{50} = 88\%$$

עבור threshold=25:

FN	FP	TN	TP
3	1	24	22

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} = \frac{22 + 24}{50} = 92\%$$

עבור threshold=35:

FN	FP	TN	TP
3	2	23	22

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} = \frac{22 + 23}{50} = 90\%$$

## ניסוי רביעי – נפילת הcador רחוק מרגלי השחקן ולצדיו

עבור threshold=15

FN	FP	TN	TP
2	1	24	23

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} = \frac{23 + 24}{50} = 94\%$$

עבור threshold=25

FN	FP	TN	TP
0	1	24	25

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} = \frac{25 + 24}{50} = 98\%$$

עבור threshold=35

FN	FP	TN	TP
0	2	23	25

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} = \frac{25 + 23}{50} = 96\%$$

## מסקנות:

כפי ששיעורנו, אכן ה-threshold האידיאלי הוא 25, מכיוון שכפי שניתן לראות הוא מושג את ה-ACC הגבוה ביותר בכל אחד מהນיסויים, ה-ACC שהתקבל היה מספיק גבוה בכל אחד מהניסויים, ולכן זה ה-threshold הנבחר.

כמו כן ניתן לראות שכפי ששיעורנו, הניסוי שהשיג את ה-ACC הגבוה ביותר הוא הניסוי הרביעי בו ברגע קפיצת הcador על הרצפה, הcador היה רחוק מהשחקן ולצדיו. הניסוי שהשיג את ה-ACC הנמוך ביותר הוא הניסוי הראשון בו ברגע קפיצת הcador על הרצפה, הcador היה קרוב לשחקן ומ לפניו.

יחד עם זאת, בפועל, מקרה בו cador נופל גם קרוב מדי לשחקן וגם מ לפניו זה מקרה קצה נדיר, מכיוון שאםcador נופל ליד הרגל של השחקן, רוב הסיכויים שהוא גם יצליח להקפי אותו.

לכן נוכל להגיד שסך הכל ביצועי האלגוריתם שלנו הם טובים עבור הבחירה של threshold=25.

## סיכום:

הפרויקט שבחרנו טמן בתוכו קשיים מוגנים איתם התמודדנו באמצעות הכלים השונים שלמדנו בקורס ובאופן פרטי. הפרויקט דרש למידה עצמאית, מאמץ וזמן רב.

המערכת המאפשרת שימוש נוח ואמין של משתמשי משחק מלך החקפות, אשר מספקת בזמן אמיתי, מידי מיטבי על כמות החקפות של השחקן כך שלא יצטרך להתעסך במלת הספירה המבלבלת תוך כדי המשחק שעלולה לגרום בהישגיו, תוך בדיקה של הקפות חוקיות, ללא מעבר על חוקי המשחק עקב נגיעה יד של השחקן בצד או נפילתו של הצד או הרצפה.

האלגוריתם הינו שילוב של שלושה אלגוריתמים עיקריים:

1. זיהוי הצד ופרט זיהוי הקפה של הצד

2. זיהוי נפילת הצד על הרצפה

3. מעקב אחר תנועת הידיים של השחקן ופსילתו עקב נגעה ביד

מצאנו שאלגוריתם זיהוי הצד עובד מצין והצלחנו לזהות את הצד כמעט בכל הפרסים

בנוסף הצלחנו לזהות הקפה של הצד בצורה מצוינת ללא חזרתיות של ספירה עבור הקפה בודדת או ספירת הקפה מיותרת

עבור אלגוריתם זיהוי הרצפה הצלחנו להפריד בצורה מאוד טובה את הרצפה מהשחקן והצד על ידי תമונות הפרשים וייצרנו אלגוריתם שאותו גם ניתן לעלה במצבים שונים שימושיים כמו כי צוין שאיתם האלגוריתם התמודד בזמן אמיתי עבור זיהוי נפילת הצד כל הרצפה

אלגוריתם מעקב הידיים עבד בצורה טובה מאוד ובמרבית הפרסים הידיים של השחקן היו מסומנות במלבן המוגדר למעקב ובאופן ספציפי האלגוריתם שהשתמשנו בו מיועד למצבים כי צוין של תנועה חדה של ידי השחקן ושינויים של מיקומי היד וזוויתיה

הפרויקט הביא אליו התמודדות עם אתגרים שונים

העובדת על הפרויקט הייתה מתוגרת בלשון המעטה. בכך המחייב בכך שהיינו צריכים ללמוד ולנסות אלגוריתמים שונים שנלמדו בכיתה וככלו שלמדנו באופן פרטי על מנת להצליח לבצע את המשימה. בכך החברתי של התמודדות עם עבודה בצוות של 4 וקביעת לוחות זמניים. מאידך העבודה הייתה מספקת ומהנה וננתנה לנו תחושת מטוגנות והישגים רבים כמהנדסים צעירים לעתיד שיכולים לספק מערכת שלמה אמינה, עובדת ובעל ערך עבור משתמשים רבים.

**ביבליוגרפיה:**

**אתר שסביר על מציאת קונטורים-**

<https://www.opencvhelp.org/tutorials/image-analysis/finding-drawing-contours/>

**-Lecture-3 HSV**

[https://moodle.bgu.ac.il/moodle/pluginfile.php/4325488/mod\\_resource/content/1/Class3-Pyramids-Color-Transformation.pdf](https://moodle.bgu.ac.il/moodle/pluginfile.php/4325488/mod_resource/content/1/Class3-Pyramids-Color-Transformation.pdf)

**פעולות מורפולוגיות-**

[/https://www.geeksforgeeks.org/python-opencv-morphological-operations/](https://www.geeksforgeeks.org/python-opencv-morphological-operations/)

**הסבר על CSRT**

<https://medium.com/@khwabkalra1/object-tracking-2fe4127e58bf>

**תנאים פיזיקליים**

חישוב מהירות הcador

חישוב תאוצת הcador

**תמונה הפרשים**

[https://en.wikipedia.org/wiki/Image\\_subtraction#:~:text=Image%20subtraction%20or%20pixel%20subtraction%20or%20difference%20imaging,an%20a%20new%20image%20generated%20from%20the%20result](https://en.wikipedia.org/wiki/Image_subtraction#:~:text=Image%20subtraction%20or%20pixel%20subtraction%20or%20difference%20imaging,an%20a%20new%20image%20generated%20from%20the%20result)

**confusion matrix**

[https://moodle.bgu.ac.il/moodle/pluginfile.php/4351020/mod\\_resource/content/1/Class7-hough-descriptors.pdf](https://moodle.bgu.ac.il/moodle/pluginfile.php/4351020/mod_resource/content/1/Class7-hough-descriptors.pdf)