



**אוניברסיטת בן-גוריון בנגב**

Ben-Gurion University of the Negev

הפקולטה למדעי ההנדסה

המחלקה להנדסת חשמל ומחשבים

Faculty of Engineering Science

Dept. of Electrical and Computer Engineering

**פרויקט גמר – מבוא לעיבוד ספרתי של תמונות**

**Blind Poker – Texas Hold'em table recognition by image processing**



ליאור בסה	203318373
ניר לוי	203528153
אסף גולדברג	305495525
עמית רוסק	203954771

**תובן עניינים:**

3.....	הקדמה
3.....	תיאור הפרויקט
4.....	הגדרת הבעיה
4.....	הנחות
5.....	מגבילות
6.....	אלגוריתם כללי
7.....	אלגוריתם זיהוי ציפים
9.....	תוצאות ומסקנות
10.....	בעיית זיהוי קלפים ודרכי התמודדות
14.....	רשימת פונקציות
15.....	תמונות המערכת
16.....	נספחים

## **הקדמה:**

טקסס הולדים (Texas Hold'em) הוא משחק קלפים השיך לקבוצת משחקי פוקר. כיוון שtexass holdem הוא משחק פוקר ששחקנים בклפי קהילה, ישנו מספר קלפים המשותפים לכל השחקנים. קלפים אלה מוצגים לכל המשתתפים עם הפנים למעלה. בתחילת המשחק מקבל כל שחקן צמד קלפים אישיים (פני הקלפים למטה) ולאחריו נערך סבב הימורים ופרישות. לאחר שנסגר ההימור הראשוני נפתחים שלושה קלפי קהילה המשותפים לכלם, הנקראים "פלופ (Flop)". מטרת השחקנים ליצור בעזרת שני הקלפים שבידם יחד עם קלפי הקהילה את היד החזקה ביותר, על פי כללי הפoker הרגילים.

לאחר שקלפי הקהילה הוצגו בפני השחקנים, נערך סבב הימורים ופרישות נוספת, ולאחר מכן נגלה קלף קהילה נוסף המכונה "טרון (Turn)" השחקנים מהמרים בשלישית בהתאם למצב החדש, ולאחר מכן מלחיטים על סכום הימור נגלה לעיני כל הקלף האחרון, המכונה "ריבר (River)" השחקנים מהמרים הימור סופי, ומגלים את הקלפים.

המנצח במשחק הוא השחקן עם היד החזקה ביותר או האחרון שנשאר אחראי שכולם פרשו בהימור. היד החזקה ביותר היא היד שהשחקן שילב בין קלפיו לבין קלפי הקהילה. הקומבינציה בין קלפי היד ובין קלפי הקהילה יכולה להיות שימוש בחלק יד אחד עם ארבעה קלפי קהילה, או שימוש בשני קלפי יד ושלושה קלפי קהילה, או שימוש בכל חמשת קלפי הקהילה כהגדרת היד שלב.

בדומה לפוקר, נהוג לשחק מספר משחקי טקסס הולדים ברציפות - כל משחק מכונה "סיבוב".  
כללי המשחק המפורטים מצורפים בסוף לדוח זה.

## **תיאור הפרויקט:**

פרויקט זה עוסק בנושא עיבוד תמורה מתוך פרוייקט כולל שמטרתו להנגיש את המשחק לעיוורים ולאנשים בעלי מוגבלות ראייה.

מטרת פרוייקט זה היא לוזות את כל האובייקטים הרלוונטיים בזמן אמת למשחק טקסס הולדים באמצעות כלים מתחום עיבוד תמורה וניתוח הנתונים כך שנוכל להמירם לפוקודות קוליות ולהשmuע אותן לאדם עיוור כך שיוכל לשחק במשחק. בנוסף, הצגת הנתונים על מסך לכל הצופים.  
כדי שהפרויקט הכלול באמצעות ינגיש את המשחק לעיוורים נוספים פרוייקט זה שעוסק בעיקר בעיבוד התמורה אמצעים נוספים כגון: צייפים עם כתב בריל, אוזניות לכל שחקן כדי שיוכל לשמעו איזה קלפים יש לו ועוד מידע שהוא בעליה לשחקן זה ולא לכל השחקנים עוד.

זהו ממחשב של שולחן משחק הינה מושלמת אשר כוללת אתגרים רבים אשר עברונו לבני אדם נתפסים כפעולות ברורות מלאיהן.

בפרויקט זה עוסוק בبنית מערכת ממוחשבת לניהול המשחק בזמן אמת, בקרה על פעולות השחקנים ועל סכומי הימור לפי חוקי המשחק, חישוב סיכויי הצלחה לכל שחקן, הצגת הנתונים על גבי מסך והקרה של מצבים מסוימים תוך שימוש בפוקודות קוליות.

המערכת עצמאית ורצה לאורך כל המשחק ללא התערבות ובקרה אנושית, מבוססת על עיבוד תמורה וכוללת אלגוריתמים רבים במטרה לוזות את כל הפרטיהם הרלוונטיים השונים בתמורה - זהו קלפים והפרדותם לקלפים אישיים או לקלפי קהילה, צייפים (איסמוני משחק המדמים כסף משחק), זהו יד (במטרה לוזות סיום תור) ואובייקטים אחרים נוספים.

באמצעות מצלמה הממוקמת מתחת לשולחן שקוֹף העשוֹי פרספקטָס אנו מסריטים את המשחק ובכל פרק זמן מסוימים נלקחת תמונה אשר אותה (או רצף תמונות כאלו) אנו מעבדים וعلاה מבצעים מניפולציות מסויימות במטרה לקבל את המידע הרצוי.

לאחר עיבוד המידע וניתוח הנקודות הכספיים יכולים לראות את המידע הרלוונטי על מסך והשחקנים יכולים לשימוש בפקודות קוליות את מצב המשחק.

### הגדרת הבעיה:

#### 1. סגןנטציה: זיהוי אובייקטים בתמונה-

- **זיהוי איזורי משחק**: איזור אישי לכל שחקן (הכולל את הקלפים האישיים שלו, הציפורים ומקום להנחת יד כסימון לסיום תור), איזור כללי להימורים (הקופה), איזור לקלפי הקהילה (Flop, Turn, River).
- **זיהוי קלפים**.
- **זיהוי ציפים**: צ'יפ מיוחד, Dealer, אשר באמצעותו נקבע מאיפה מתחילה סיבוב משחק וצ'יפים רגילים במצבים שונים אשר מדמים כסף משחק בשווי שונה לפי צבע.
- **זיהוי יד**.

#### 2. עיבוד נתונים ובקרה בזמן אמיתי: בדיקת חוקיות הפעולות של שחקן (Fold, Check, Call, Raise) בקרה על כמות הציפורים שהוא מוסיף לקופה הכללית, חישוב סיכוי הצלחה לכל שחקן, השמעת פקודות קוליות מתאימות, הצגת הנקודות על מסך ואimotoת סוף תור של שחקן בודד, סיבוב הימורים, או משחקו מלא (כאשר שחקן אחד או יותר מניצחים).

\* כל אחת מהתביעות הללו נפתרת באמצעות אלגוריתמים שונים המחלקים את התביעות הללו לתתי בעיות נוספות.

### הנחות:

- שולחן שקוֹף העשוֹי פרספקטָס (עם סימונים של חלוקה לאיזורים) כדי שהמצלמה תמוקם מתחת לשולחן.
- התקירה של החדר בו משחקים בצד לבן.
- השולחן מחולק לאיזורי משחק- איזור אישי לכל שחקן ובו מקום לקלפים, לצ'יפים, לצ'יפ Dealer, ולהנחת יד, איזור מוקצה לקלפים המשותפים, איזור הקופה (כל השולחן למעט האיזורים האחרים שהוגדרו).
- כל הקלפים הפיכים, גם הקלפים האישיים וגם הקלפים המשותפים (בשונה ממשחק רגיל בו הקלפים המשותפים מוצגים לכל השחקנים).
- פתיחת Flop מכיוון מסוימים.
- הקלפים מסוימים לצורך זיהוי (פירוט בהמשך הדוח').
- אין חפיפה מלאה של צ'יפים באיזור הקופה כדי שנוכל לזהות אותם.
- הנחת יד לצורך סימון סוף תור מתבצעת כשהאכזבות מופרדות ולא צמודות ובמקומות המסומן כאשר לא מונחים שם אובייקטים נוספים (כגון צ'יפים).

#### **מגבילות:**

- מצלמה ייחודית- במידה והיו לרשותינו מספר גדול יותר של מצלמות האלגוריתם היה יציב יותר והנחה מסוימות שביצעו לא היו נדרשות, לדוגמה, ההנחה שגם הקלפים המשותפים הפוכים או סימנו הקלפים לצורך זיהוי. העובדה שאנו משתמשים במכשיר ייחידה גורם לכך שלאחר בידוד איזורי התמונה וחיתוך התמונה בהתאם הרזולוציה ואיכות התמונה נפגעות משמעותית ופוגעת ביכולות הזיהוי של האלגוריתם.
- תאורה- תנאי התאורה בחדר משפיעים על פרמטרים מסוימים שבאמצעותם אנחנו מסוגים אובייקטים, לדוגמה, סיוג אובייקט באמצעות צבע (שינוי בהירות התמונה ובהתאם שינוי ערכי הצבע המינימליים והמקסימליים כתנאי לסיוג אובייקט כלשהו).
- שולחן שקוף- מכיוון שהשולחן שקוף נוצר מצב שם נוספים אובייקטים מעבר לאביזרי המשחק האלגוריתם עלול לזהות אובייקטים לא רצויים שייפגעו בעיבוד הנתונים (תקלה לא חלקה, שחקון שרוכן מעל השולחן וליבש חולצה עם צירורים כלשהם ועוד).
- MATLAB- במידה והיתה לנו אפשרות לעבוד עם תוכנות או שפות תכנות אחרות כגון ספריית OpenCV היו עומדים לרשותינו כלים נוספים לשיפור האלגוריתמים (Deep Learning) ועוד).

### **אלגוריתם כללי:**

1. הסיטה באמצעות המצלמה כך שהמצלמה תופסת את כל השולחן.
2. לكيחת תמונה בודדת מהסיטוון כל פרק זמן מסוים (זמן משתנה בהתאם ליריצת הקוד).
3. עבר כל תמונה- זיהוי האיזורים השונים בשולחן וחישוב התמונה בהתאם.
4. בכל שלב במשחק מוצעות הפעולות הבאות :
  - זיהוי צייף Dealer וקביעת התור של השחקן הראשון בכל סיבוב הימורים
  - זיהוי קלפים אישיים ומשותפים
  - חישוב סיכויי הצלחה לכל שחקן
  - זיהוי ציפים
  - בדיקת הימורים- Small / Big Blind, השוואת אליהם או להימור הגובה הנוכחי, הعلاה, פרישה מהסיבוב או צ'יק.
  - עדכון סכום הכספי שנשאר לכל שחקן
  - זיהוי יד למשך 2 שניות על השולחן לטימון סוף תור של שחקן

שלבי המשחק :

שלב 0- תחילת המשחק, חלוקת קלפים אישיים

שלב 1- פטירת Flop

שלב 2- פטירת Turn

שלב 3- פטירת River

שלב 4- סיום משחקנו וקביעת מנצח או מנצחים.

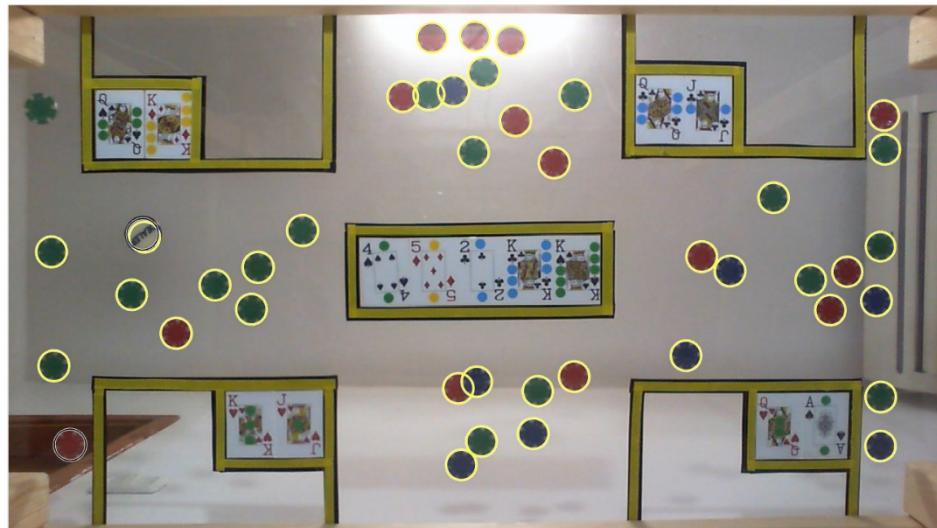
בפרויקט זה מושימות מרובות ולכל אחת מהן אלגוריתם משלها, נציג בקצרה את כל הפונקציות בהמשך ונתעמק באחד מהאלגוריתמים כולל ניתוח תוצאות ומסקנות.

בנוסף, בחרנו קושי מהותי שהיה לנו בפרויקט עקב המוגבלות, ננתח את הבעיה ודרך הפתרונו שהצענו להתמודדות עם הבעיה.

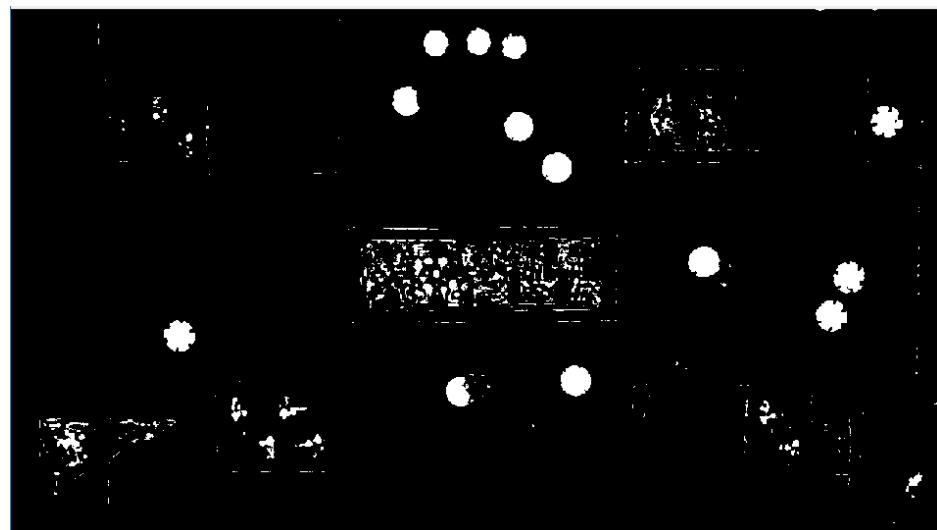
### אלגוריתם זיהוי צ'יפס:

האלגוריתם זיהוי צ'יפס עובד בעיקר על שתי שיטות סינון עיקריות:

- מציאת עיגולים עיי שימוש ב Hough-circles



- סינון צבעים בעזרת HSV (getchipcolor)



בשלב הראשון אנו מוחפשים את כל העיגולים באמצעות הfonkczie  
imfindcircles(gray\_table,[18 22],'ObjectPolarity','dark','Sensitivity',0.97,'EdgeThreshold',0.1)  
כאשר הפרמטרים השונים ממשמעותיים מאוד לזיהוי.

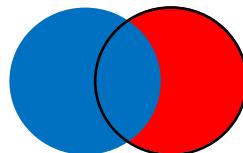
לאחר מכן, אנו מבצעים סינון של כל העיגולים לפי 2 דרכי:

1. רדיוס העיגול – לאחר והצלמה נמצאת במרקם קבוע מהפרשפאס אנו יודעים בדיקת טווח הערכים.

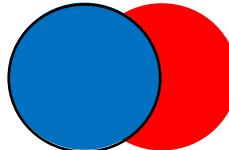
2. מיקום מרכז העיגול – בדיקה האם מרכז העיגול נמצא באיזור המוגדר כקופה ולא באיזוריים האישיים.

לאחר הסינון הנ"ל אנו בודקים מה הצבע הממוצע של הפיקסלים הנמצאים בתוך העיגול, לפי זה ניתן להחליט האם העיגול הוא צייפ וממה השווי שלו, אדום שווה 1, ירוק שווה 2 וכחול שווה שלוש. האלגוריתם המתואר לעיל עובד על המצב ה"קלاسي", כלומר, מצב בו כל הצ'יפים מונחים על השולחן ללא חיפוי (אין צ'יפים שעולים אחד על השני).

במצבים מורכבים יותר בהם קיימת חיפוי בין צ'יפים, אם ננסה למצוא את הצבע הממוצע בתוך העיגול נקבל ערכים חלקיים ולכן האלגוריתם לעיל לא יעבוד, במצב כזה אנחנו משתמשים בפונקציה עזר הנקראת chipdecision. בפונקציה זו ביצענו רקורסיה הבודקת את כל הצ'יפים החופפים, במידה יש חיפוי של יותר משני צ'יפים, המעבר לצ'יף הבא נקבע לפי המרחק הקרוב ביותר של מרכז המעגל (הצ'יף) החופף, ולבסוף מתקובלת החלטה לגבי הצבע הדומיננטי, לדוגמה :



כאשר נחפש את הצבע בתוך העיגול השחור אנחנו לא נדע להחליט אם זה הצ'יף הכחול או האדום. במצב הנ"ל, אנו נחפש את הצ'יף הקרוב אליו (זה שחותף אותו) :

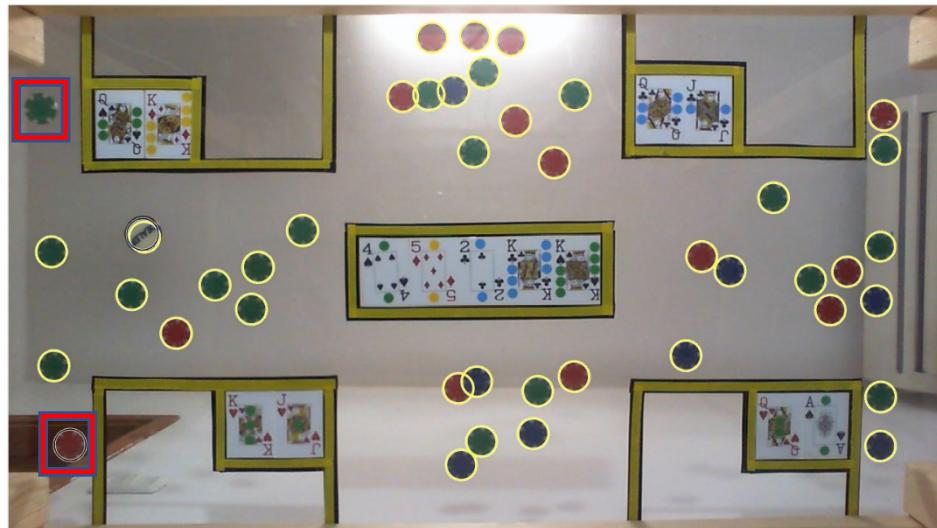


cut, אנו נדע לקבוע כי הצ'יףצבע כחול (בעיגול השחור), לעומת זאת התלבטו מקודם בין כחול לאדום אז קיבל החלטה כי הצ'יף הקודם היה אדום. האלגוריתם עובד גם עבור חיפוי של יותר משני צ'יפים.

## תוצאות ומסקנות:

כפי שניתן לראות בתמונה לעיל יש אחזוי הצלחה גבוהה לאלגוריתם, אמנם, קיימות מספר סיבות לכישלון:  
1. תנאי התאורה משפיעים בצורה משמעותית על התוצאות ולכן צריך להתאים פרמטרים מסוימים לתנאי תאורה משתנים.

בנוסף, מאותה סיבה יכול להיווצר מצב בו ציפים מסוימים לא יזוהו כי איזורים שונים בשולחן מוארים בעוצמה שונה, נסמן אותם בריבוע אדום עבר התוצאה המוצגת.



לכן, כדי למנוע מצבים אלו אנו מניחים ששחקן לא יניח ציפים בקצוות השולחן בהם האיזורים חשובים יותר (בהתאם לתנאי התאורה).

2. מצב נוסף בו הזיהוי יכול להיכשל הוא מצב בו יש חיפוי גדולה משמעותית אשר קרובה לחיפוי מלאה. סיבה ראשונה לכישלון- הfonkcija שבאמצעותה אנו מוצאים את העיגולים, imfindcircle, נכשלה ואנו מוצאים עיגול ייחיד כיון שאחزو החיפוי גדול מאוד (מצב בו אנו קרובים מאוד לחיפוי מלאה). סיבה שנייה לכישלון- המיצוע בין שני צבעים יחזיר לנו ערך של הציף הלא רצוי כיון שהוא מכסה את רוב השטח של הציף השני, אך נניח שחיפוי הגודלה מחצי ציף מועדת לכישלון. קשה להגדיר מה הגבול המדויק בו לא נצליח לזהות ציף בעקבות חיפוי מכיוון שהמיצוע של הצבעים משתנה בהתאם לתאורה, לצורך העניין, בדיקה על זוג ציפים עם חיפוי זהה לזוג ציפים אחר אך בתנאי תאורה שונים עלולה להוביל לתוצאות שונות ולכן קשה לומר באופן כללי מה הגבול אך הסתברותיתacha אחזוי הצלחה גבוהה.

3. מצב נוסף בו הזיהוי יכול להיכשל הוא מצב בו הרקע של התמונה (לרוב התקופה כי אנחנו מצלמים מתחת לשולחן כלפי מעלה או השתקפות מהרכפה כתוצאה מהארה על הפרספקט שמתפרק כਮזריר או) לא בהיר או שקיימות הפרעות שונות, למשל שורוכן מעל השולחן ולבוש חולצה בצבעים מסוימים.

זיהוי אובייקטים נוספים עלול להביא למצב בו אנו מזוהים אובייקט שהוא לא ציף אך עונה על כל התנאים והמגבלות שהזכרנו לעיל וכן מזוהה כציף שקרי. כדי למנוע מצב זה אנו מניחים שהרכפה כהה והתקפה בהירה וכן מבצעים בדיקות רבות על תמונות שונות.

### **בעית זיהוי הקלפים ודרך התחמיזות:**

אחת הפעולות המשמעותיות בפרויקט זה היא זיהוי קלפים.

כאמור, אנו משתמשים בצלמה יחידה אשר מצלמת את כל השולחן, כדי לעשות הפרדה בין קלפים אישיים ולהתאים לכל שחקן וגם לזהות קלפים מסווגים אנו מחלקים את השולחן לאיזורי משחק. באמצעות סימוניים על השולחן אנו יודעים לזהות איזוריים אלו ובהתאם לכך אנחנו חותכים את התמונה לתתי תМОונות, פעולה זו פוגעת ממשמעותית ברזולוציה ובאיכות התמונה.

לכן נוצר מצב שכדי לזהות קלפים ניגשנו לבעה באמצעות 5 גישות שונות, ארבעת הגישות הראשונות שניסינו הצליחו בסופו של דבר בגל מגבלה זו. בנוסף, בלי קשר לגישות אלו ניסינו לחזור את התמונה בדרכים שונות, לנסות לשפר את התמונה לאחר החיתוך ועוד, ללא הצלחה.

חשוב לציין שהחלק מגישות אלו נבדק עבור מאגר תМОונת שייצרנו לבדיקות (ולא ניתן להפיק את אותה האיכות כמו של מאגר זה באמצעות המצלמה הבודדת בה אנו משתמשים) וקיבלו 100% הצלחה. בנוסף, במידה והיינו יכולים להשתמש במערך מצלים (מצלמה לכל שחקן ואחת נוספת לקלפים המשתתפים) או בצלמה עם איקות גבוהה בהרבה מהמצלמה שלנו האלגוריתמים המוצעים היו מצליחים.

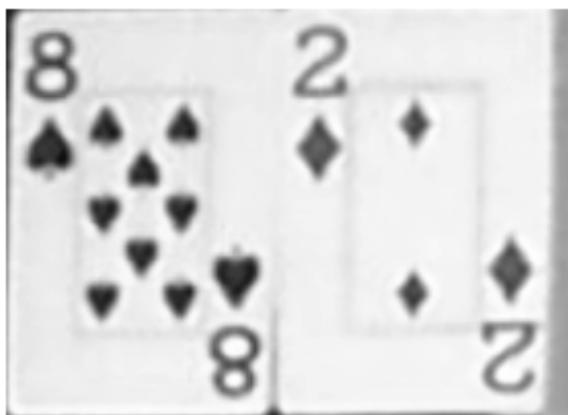
נציג את הגישות המוצעות:

דרך פתרון 1 :

זיהוי איזורי משחק באמצעות פונקציית ('ax, 'BoundingBoxBW, regionprops) אשר מקבלת תמונה בינהירות וסוג מאפיין ויודעת להחזיר את כל המאפיינים בתמונה זו, במקרה שלנו בחרכנו איזוריים מתוחמים מרובעים ובאמצעות סינון של מקומי פיקסלים, אורך ורוחב של איזוריים אלו בתמונה ידענו לבודד את האיזוריים האישיים של כל שחקן ואת איזור ה- Flop.

**תמונה של regionprops**

כיוון שהאיזוריים הללו יתקבלו בגודל קבוע בכל תמונה ידענו לחזור מתוכם את הקלפים באמצעות חיתוך באורך ורוחב קבועים בהתאם.



לאחר מכון חתכנו מותוך הקלף באותו אופן, גם הוא התקבל בגודל קבוע, את הפינה השמאלית העליונה ושוב באותו אופן השתמשו בפונקציית ('ax, 'BoundingBoxBW, regionprops) כדי לאתר את המספר והצורה של קלף זה.

לסיום, ניסינו להשתמש בשיטות OCR (OCR) אשר מקבלת תמונה כקלט ומחזירה את הטקסט שהיא מזהה בתמונה כפלט. מכיוון שאיקות התמונה הייתה ירודה הפונקציה לא הצליחה לזהות טקסט בכל הפעם והאלגוריתם לא היה יציב.

לכן ניסינו להעביר פילטרים מסוימים על התמונה, כגון פילטר גאוסי או חציוון, ניסינו להשתמש בפונקציות שמבצעות *deblurring* (דבלריזציה). עוד אך כל אלה כשלו. דוגמאות לפינוט חתומות של קלפים:



#### דרך פתרון 2:

דרך פתרון זו ייחסית פשוטה ולא אופטימלית מבחינה זמן ריצה ומורכבות זכרון. ייצרנו מאגר של 52 קלפי משחק שצלמנו בעצמו בתנאים כמה שיותר אופטימליים (ללא השתקפות או, מצלמה באיכות גבוהה ועוד), לאחר בידוד של הקלפים מהתמונה הכלולת בדרך המתוארת לעיל השתמשנו בפונקציה (I) *normxcorr2*(Template, I) העובר כל קלף שבودנו מהתמונה לכל אחד מ-52 הקלפים במאגר, כלומר *ביצוע קורלציה*. הפלט בפונקציה זו הוא מטריצת מקדמי הקורלציה בערכיהם בתחום [-1, 1], כאשר הערך המקסימלי מייצג את הפיקסל שהסבירה שלו יוצרה את ההתאמנה הגבוהה ביותר. מכאן שקיבלונו 52 מטריצות אלו, מצאנו את הערך המקסימלי מכל מטריצת וייצרנו מערך של 52 ערכי מקסימום, לקחנו מערך זה את הגובה ביותר וציפינו לקבל לפי האינדקס של ערך זה במערך את הקלף המתאים. שיטה זו לא הצליחה גם כאשר ייצרנו מאגרי קלפים שונים, למשל, ייצרנו מאגר של תמונות מהצלמה בה אנחנו משתמשים, מאותו מרחק ובאותם תנאים תאוריה שאנו עובדים איתם. ניסינו לבודד בשיטה זו גם עבור פינה שמליתعلינה של הקלף ולבצע קורלציה במקום ל-52 קלפים ל-17 סימboleים (9 ספרות, אס, מלך, מלכה, נסיך, תלתן, לב, יהלום ועליה).

#### דרך פתרון 3:

בדרכ פתרון זו גם יש שימוש במאגר תמונות ולכן לא אופטימלי, השתמשנו במאגרים שונים וניסיונות השוואת שונים. הפעם ניסינו לזרות Features בתמונה ולהשווות אותם ל- *Features* של 52 הקלפים מהמאגר. השתמשנו בפונקציות הבאות:

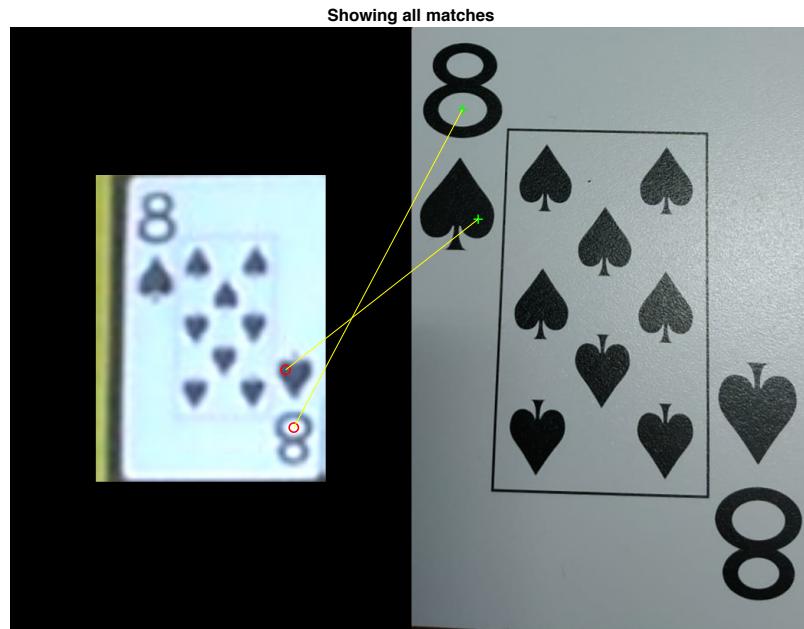
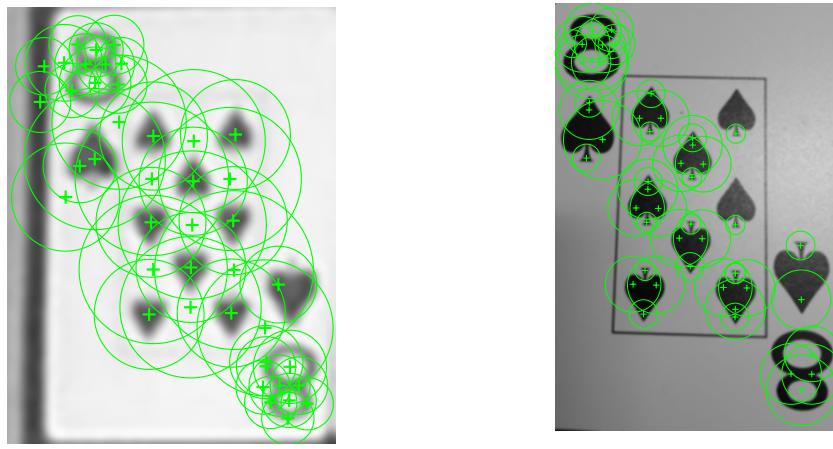
(I,2) – *imgaussfilt* – פילטר גאוסי

(I) – *detectSURFFeatures* – זיהוי Surf Features

– הוצאת נקודות עניין (לפי הסביבה של נקודה) *extractFeatures(I, I\_pts)*

– התאמת בין האובייקטים שנמצאו ב-2 תמונות *matchFeatures(ref\_features, I\_features)*

מכיוון שאיכות התמונה ירודה לא הצליחנו למצוא אובייקטים יציבים ורלוונטיים כדי שנוכל להשוות אותם לאובייקטים שנמצאו בתמונות המאגר.



גם הפעם, ביצעו השוואה מול מספר מאגרים שונים.  
כאשר בדקנו מול מאגר התמונות שייצרנו באמצעות המצלמה (מאגר תמונות באיכות ירודה) קיבלנו 93.7% אחוזי הצלחה (15 הצלחות מתוך 16 נסיעות) אך הזמן היה איטי והפתרו לא היה יציב, הספיקו שינויים קטנים בתמונות חדשות כדי שאלגוריתם זה ייכשל.

#### דרך פתרון 4:

באמצעות פונקציית ('canny edges(I,' נוציאו את הקצוט של התמונה, לאחר מכן העברנו את התוצאה בפילטרים שונים ולבסוף ניסינו למלא את הקצוט שמצאנו ובכך למנוע את בעיית האיכות הירודה עיי' ייצור תמונה בינארית. לאחר מכן רצינו להשתמש באחת מהשיטות לעיל, למשל ocr או קורלציה למאגר תמונות ולמצוא את הקלף. הבעיה בדרך פתרון זו היא שלא תמיד הצלחנו להוציא קצוט מסויק ברורים ואיכותיים כך שנוכל למלא אותם ולקבל את התוצאה הרצויה.

#### דרך פתרונו 5 :

בדרך פתרון זו השתמשנו בהנחה מוקלה עברינו והדבקנו מדבקות עגולות בצבעים שונים על הקלפים. את כל קלפי המספרים (10-1) אנו מזיהים את ערך המספר ע"י זיהוי מספר האלמנטים הנמצאים במרכז התמונה באמצעות פונקציית `regionprops`.

האלמנטים הרצויים בתמונה נמצאו לפי המיקום במרכז התמונה והגבלת רוחב המסגרת. את צורת הקלף אנו מזיהים ע"י שתי מדבקות (בראש ובתחתית הקלף) שהדבקנו, את צבע המדבקה זיהנו ע"י סינון HSV.

מדבקה ירוקה מסמלת עליה, מדבקה צהובה מסמלת יהלום, מדבקה כחולה מסמלת תלtan וקלף ללא מדבקה הינו לב.

את קלפי המלוכה סימנו ע"י מדבקות בשתי צידי הקלף : אחת בכל צד עברו נסיך, שתי מדבקות עברו מלכה ושלוש מדבקות עברו מלך.

צבע המדבקות משתנה בין צורות הקלפים לפי הסימן שצוין לעיל.

את קלפי המלוכה מסוג לב סימנו בעזרת מדבקות ירוקות במרכז הקלף (מדבקה אחד שתים ושלוש בהתאם לבן המלוכה).

### **רשימת פונקציות:**

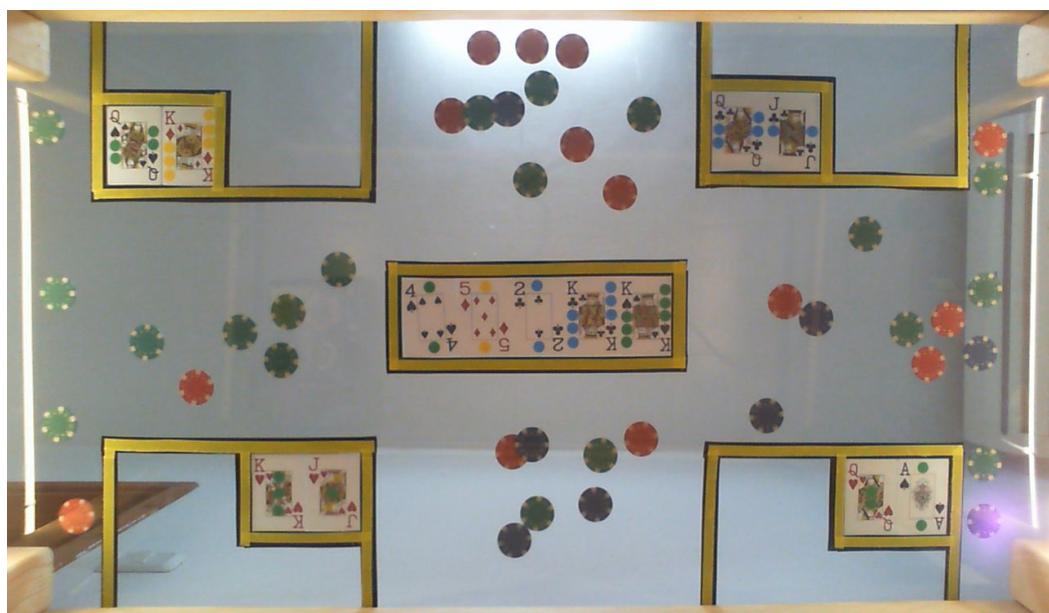
- פונקציות למשך המשחק ועיבוד הנתונים בזמן אמת :
    - Gameon – הפונקציה הראשית ממנה מרכיבים את הקוד ומכללה את כל התנאים למשך משחק תקין
    - Winning – בודק מצב של ניצחון בסוף סיבוב עם כל האפשרויות (שוויון בין מס' שחקנים וכו')
    - Nextplayer – פונקציה שקובעת מי השחקן הבא בתור ומתהשבת בכל האפשרויות הקיימות
    - Collectivecall – פונקציה שבודקת האם כולם השוו לסכום הגובה ביותר לקביעת סוף סיבוב הימורים
    - CardAnalyze – חישוב סיכויי הצלחה של שחקן
  
  - פונקציות סגמנטציה
    - Board\_detect – זיהוי גבולות האיזורים השונים בלוח המשחק
    - Card\_detect – זיהוי קלפים
    - Chips\_detect – זיהוי צייפים
  
  - Is\_pixel\_inside – פונקציה עזר לזיהוי צייפים, קובעת אם פיקסל מסוים בתוך הציף או לא
  - Get\_approx\_chip – פונקציה עזר לזיהוי צייפים וקורסיבי, מוצאת את הציף הקרוב ביותר לציף הנוכחי
  
  - Get\_chip\_colors – קובעת באיזה צבע הציף ובההתאם לכך מה השווי שלו
  - Chip\_decision – פונקציה רקורסיבית למציאת צייפות עם חפיפה
  - Handonboard – זיהוי יד המונחת לפחות 2 שניות לסתימון סיום תור של שחקן
- 
- פונקציות למסך המשתמש :
    - blindpokergui
    - playercardoff
    - playerbackground
    - setchipfunc
    - tts

**תמונהות המערכת:**

תמונה של השולחן השקוּ עם המצלמה ממוקמת מתחתיו במרכז



תמונה שצולמה מהמצלמה מתחת השולחן



## נספחים:

הסבר על חוקי המשחק: (מקור- <http://www.texasholdem.co.il>)

המשחק מורכב כמו כל משחק קלפים אחר מקלפי קהילה רגילים, דילר, שחקנים וכסף. המשחק מתחילה כאשר הדילר מחלק שני קלפים לכל אחד מהשחקנים, ולא מחלק את שלושת הקלפים הגולויים עד שנקבע סכום הימור מינימלי. אחרי שנקבעו וסוכם הסכום המינימלי הדילר מחלק את שלושת הקלפים הראשונים על השולחן הנקראים "פלופ" (Flop). "וכאן מתחילה הסיבוב הראשון במשחק. כל השחקנים יבדקו את הקלפים שעל השולחן עם הקלפים שיש להם ולקבוע אם השילוב של שני הקלפים שיש להם יכולם לנצח או לא (מה שנקרה יד חזקה) וכן כרגע ישן שלוש אפשרויות והן:

ליותר על המשחק ולסגת מאחר שחקן הבין שאין לו יותר סיכוי לנצח במשחק והוא עדיף לו להפסיק סכום מינימלי ששם בתחילת המשחק מלהפסיד בהמשך המשחק את כל סכום ההימורים שיכל להגיע לאלפי דולרים. אפשרות זו נקראת FOLD - האפשרות השנייה היא להפסיק במשחק לסיבוב השני ולהישאר עם הסכום המינימלי שקבעו בתחילת המשחק – זה במקרה והשחקן עדין לא בטוח שהקלפים שלו מנצחים ומוקווה שהסיבוב הבא יעוזר. מה שנקרה CHECK האפשרה האחרונה היא להפסיק במשחק ולהעלות את סכום ההימור – לכל מי שבוטח שהוא מנצח במידים אחרים RAISE השחקנים חייבים להסכים עם הסכום החדש שנקבע ע"י אותו שחקן שעשה RAISE וכן מתחילה תהליך חדש של בדיקת כדאות להישאר במשחק או לסתות ממנו לכון השחקן שיושב אחרי מי שהעלה את סכום ההימור לפי כיוון השעון הוא זה שמתחל בנסיבות שבו מחליטים להישאר או לסתות ועד שיגיעו למי שהעלה את הסכום ומcause עוברים לשלב של הסיבוב השני.

הסיבוב השני ממשיך עם השחקנים שנשארו מהסיבוב הראשון, כאן הדילר מחלק קלף ייחיד גלווי על השולחן שהוא הרביעי המכונה "טַרְן" (Turn) "וכאן מתחילה עוד תהליך שבו צריך להחליט אם להישאר, לסתות או להישאר ולהעלות את הסכום.

בסיוב השלישי הדילר שmas את הקלף החמישי והאחרון המכונה "רייבר" (River) "ומי שנשאר כאן יctrיך להחלטת אם הוא נשאר, פורש או מעלה את סכום ההימור. השחקנים שיישארו יהיו חייבים להראות את הקלפים שיש להם ולבודק מי באמת יש לו את שני הקלפים שישתלבו הכி חזק (היד החזקה ביותר) עם שלושה מהקלפים הגולויים על השולחן. מי שניצח יזכה בכל הכספיים של כל השחקנים. כל משחק הפוֹרְקְרָקְסְס הולדם כוללים סכום מינימלי וסכום מקסימלי כהימור התחלתי ועל כל אחד מהשחקנים להסכים. מי שיוציא מהמשחק בסיבוב הראשון או השני לא ישחק בסיבובים אחרים. ומה הם השילובים הכி חזקים ומנצח?! כאן תוכלם למצוא את כל אחד מהשילובים האפשריים לפוי השם שלו ומהחזק ביותר ועד לחיש ביותר-

ROYAL FLUSH- הרץ המלכותי



חייב להיות חמישה קלפים עוקבים ומאותו סוג אך החל מ 10 עד האס . זה נחשב לשילוב הכי חזק בכל המשחק.

### STRAIGHT FLUSH



שייהו חמישה קלפים עוקבים אחד את השני מאותו סוג אך מעט מ 10 עד אס כמו 6,5,4,3,2 או 8,7,6,5,4 וכו...'

### FOUR OF A KIND



4 קלפים זהים ומאותו סוג בנוסף לפחות אחר

### FULL HOUSE



צריך שיהיו שלושה קלפים זהים ושניים אחרים גם זהים

### FLUSH



5 הקלפים חייבים להיות מאותו סוג של קלפים ולא חשוב מה הסדר שלהם ...

## STRAIGHT



5 הקלפים חייבים להיות עוקבים אחד את השני ולא חשוב מאיויה סוג ...

## THREE OF A KIND



השילוב הזה מורכב משלוש קלפים זהים ושני קלפים אחרים לא חשוב מהם מספר או סוג.

## TWO PAIRS



שיינו שני זוגות של מספרים זהים כמו שני קלפי 4 ושניים 5 ואחד נוסף לא חשוב מה יהיה.

## ONE PAIR



שילוב של זוג זהה אחד ושאר הקלפים שונים ואין להם קשר.

## NO HAND poker no pair OR HIGH CARD

מספיק שייהי קלף אחד חזק – קלף מס 2 הוא הכי חלש וקלף האט הוא הכי חזק.

נדיר מאד להגיע לשילוב הראשון, החזקה ביותר, لكن במקרה ויש את השילוב האחרון  
עדיף להתקפל ולסגת מהמשחק FOLD.