

מעבדה לעיבוד אותות פיזיולוגיים

דו"ח מסכם - עיבוד תמונה

מגישי הדו"ח המזהים:

ארן אהרון	עדן קורש	אוראל בטיטו
209010156	206845315	315432724

mdiricht_hanotah_merennah:

nofer_yirmi

26/01/2024

	1	תוכן עניינים
3.....	2	תקציר
4.....	3	ניסויים
4.....	3.1	ניסוי 1
4.....	3.1.1	היפותזה.....
4.....	3.1.2	מתודולוגיה.....
4.....	3.1.3	תוצאות.....
10.....	3.1.4	מסקנות.....
11.....	3.2	ניסוי 2
11.....	3.2.1	היפותזה.....
11.....	3.2.2	מתודולוגיה.....
11.....	3.2.3	תוצאות.....
14.....	3.2.4	מסקנות.....
15.....	3.3	ניסוי 3
15.....	3.3.1	היפותזה.....
15.....	3.3.2	מתודולוגיה.....
15.....	3.3.3	תוצאות.....
22.....	3.3.4	מסקנות.....
24.....	3.4	ניסוי 4
24.....	3.4.1	היפותזה.....
24.....	3.4.2	מתודולוגיה.....
25.....	3.4.3	תוצאות.....
31.....	3.4.4	מסקנות.....
32.....	4	מסקנות כלליות.....
33.....	5	ביבליוגרפיה.....
34.....	6	נספחים.....
34.....	6.1	נספח א - קוד המطلב

2 תקציר

במעבדה זו חקרנו את מבנה התמונה ותכונותיה במישור המרחבי ובתחום התדר, תוך התמקדות בסינון תМОנות, ביצוע פעולות מורפולוגיות, סגמנטציה, זיהוי גבולות, וחילוץ מאפייני אובייקטים. כמו כן, ניסינו לעבד ולנתח סרטוני וידאו. המעבדה העמיקה את הבנתו בתחום עיבוד תמונה וויזואו לזיהוי וניתוח של תכונות ואובייקטים ויזואליים, תוך הבנת חשיבות השימוש במסכות נוכנות, השרת רעשים, והתאמאה בין תМОנות כיול לויזיאו. כמו כן, הדגשנו את הצורך בשימוש לצורך איקוטי ובטכניקות מתקדמות לקבלת תוצאות מדויקות יותר.

בניסוי הראשון, עבדנו על תמונה א/orו אליה התייחסנו כתסיות דם, כשמטרתנו הייתה לבצע סינון כך שנקבל תמונה עם הטסיות הנוטות ימינה בלבד. השרת הרקע הובילה לשיפור באיכות הסגמנטציה ואפשרה ייצור מסכה מדויקת יותר. כמו כן, השתמשנו בפילטרים ושיטות שונות לזיהוי גבולות, והדגמנו את הייעילות של שימוש במאפיינים כמו אוריאינטציה לזיהוי והפרדה של אובייקטים לפי כיווניות מסוימת.

בניסוי השני, סיננו רוש מלך פפל מתמונה באמצעות מסנן חציון, וקיבלו כי גרעין בגודל 3x3 הינו האופטימלי לסינון הרוש תוך כדי שמירה על חדות התמונה עבור התמונה שסיננו.

בניסוי השלישי, סיוגנו כיווני הסטכלות של נבדק בסרטון וידאו, תוך שימוש בתמונות כיוול ומציאות מיקומי העניינים. התאמה נכונה לתמונות הכוול הוכיחה את חשיבותה לאחיזה ההצלחה של זיהוי אוור הסטכלות. בנוסף, ביצוע מבחנים סטטיסטיים יובילו לכימיות טוב של טיב הזיהוי וישנה חשיבות רבה לבחירת המבחן הסטטיסטי המתאים.

בניסוי הרביעי, ניתחנו את תנומת רגלו הימנית של הנבדק באמצעות מרקרים, תוך ניתוח מיקום מרקרים, מהירות, וזריות הרגלים. ראיינו את החשיבות של איקות המצלמה, יציבותה, ותאורה מתאימה לזיהוי מדויק של המרקרים. הסקנו כי דרך זיהוי המרקרים אינה מספקת ונדרש ליעיל את שיטת הזיהוי מעבר.

3 ניסויים

3.1 ניסוי 1

3.1.1 היפתוזה

בניסוי זה השתמש בתמונה של אורו, אליה נתייחס במהלך הניסוי כתמונה של טסיות דם. מטרת הניסוי היא ליצור תמונה המורכבת אך ורק מטסיות הנוטות ימינה (ז) ולא הנוטות שמאליה (א), ככלمر להישאר רק עם כחץ מה特斯יות בתמונה. לשם כך אנו נדרשים לבצע סגמנטציה ולאחר מכן מיצוי מאפיינים של טסיות הדם. את המסכה ניצור פעמיים אחת בעזרת התמונה המקורית ופעמיים בעזרת התמונה המקורית ללא הרקע אשר נשעך באמצעות erosion¹. נצפה כי בשתי הדריכים נצליח להבדיל בין הטסיות לרקע, זאת בשל ההבדלים הבנוראים יחסית הקיימים ביןיהם ובאפשרים בחירת סף טוב מספיק. כמו כן, נצפה כי המסכה המיטיבית להפרדת טסיות מהרקע התקבלה מהתמונה ללא הרקע, זאת מכיוון שהחכרתו התקבלה תמונה עם רמת רעש נמוכה יותר. לאחר מכן נבצע זיהוי גבולות בעזרת שתי שיטות, שיטת LoG ושיטת dilation² והחכרת המסכה. נצפה כי בשתי השיטות יתקבלו תוצאות טובות וכי נצליח להבדיל בצורה טובה בין הטסיות הפוניות ימינה לאלו הפוניות שמאליה בשל האוריינטציה השונה שלן וכן לקבל תמונה סופית המציגת את הטסיות הפוניות ימינה.

3.1.2 מתודולוגיה

כלים וחומרים: תוכנת המטלב.

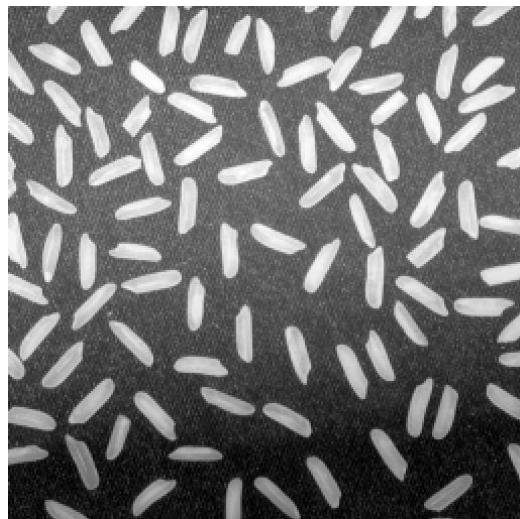
מהלך הניסוי: ראשית, נטען את קובץ התמונה המקורי במטלב. לאחר מכן נציג את רמות האפור של התמונה בעזרת היסטוגרמה, נסחה למצוא סף ייחיד המבדיל בין הטסיות לבין הרקע ובעזרתו נבנה את המסכה. ניצור מבנה בצורת דיסק בעזרת פקודה strel במטלב בגודל שנקבע בעצמינו וכן נעלים את הטסיות לקבלת הרקע בלבד. בשלב השני, ניעזר בפקודות bwlabel ו-regionprops על מנת למצוא רק את הטסיות הנוטות ימינה וניצור מסכה המכילה רק אותן.

3.1.3 תוצאות

תשובות לשאלת 1.1:

נציג את התמונה המקורית:

Blood Platelets Photo

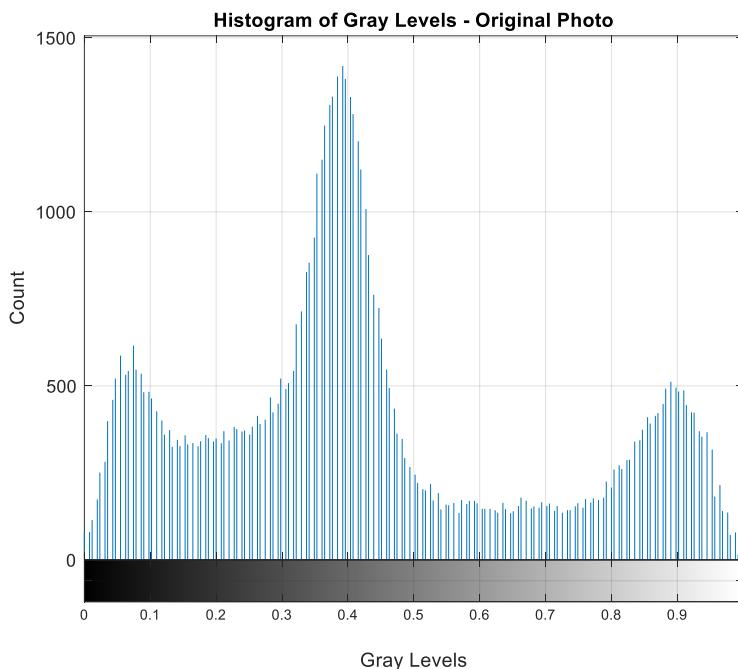


אייר 1 : תמונה טסיות הדם (אורו)

באייר 1 ניתן לראות את תמונה טסיות הדם עליה נבצע את הפעולות. נשים לב כי גוון הטסיות בהיר יותר מרקע התמונה. כמו כן, רקע התמונה אינו אחד, בחלוקת התמונה של התמונה הרקע השחור יותר (ככל הנראה צל) ובחלוקת העליון בהיר יותר.

תשובה לשאלה 1.2 :

מציג את היסטוגרמת רמות האפור של התמונה לעיל:

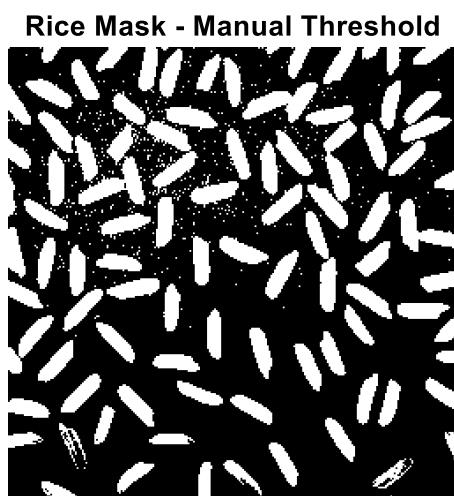


אייר 2 : היסטוגרמת רמות אפור

באייר 2 ניתן לראות את היסטוגרמת רמות האפור של תמונה הטעיות. נשים לב כי ישנו 3 פיקים משמעותיים ; בסביבות 0.9 - ככל הנראה אלו גווני טסיות הדם (בהיר), 0.4 - ככל הנראה הרקע בחלקן העליון של התמונה ו- 0.05 - ככל הנראה הרקע בחלקו התיכון של התמונה (כהה). למרות הפיקים הנ"ל נראה כי התפלגות רמות האפור מתרפרשת על כל הטווח כולם לרמות האפור מופיעות ברמה זו או אחרת בתמונה.

תשובה לשאלה 1.3 :

בחרנו את הסף להיות 0.5 לאחר שבחנו את ההיסטוגרמה באייר 2 ואת המסקנה המתקבלת מספים שונים. להלן המסקנה הטובה ביותר שי��בנו :



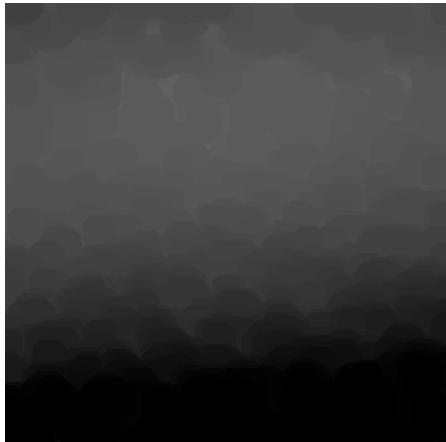
אייר 3 : המסקנה שהתקבלה בבחירה הסף להיות 0.5

באיור 3 ניתן לראות את המסכה שייצרנו. נשים לב כי רוב הטסיות מסומנות בצעע לבן במלואן חוץ מכמה טסיות בודדות מתחתית התמונה. למרות זאת, קיבל רعش של נקודות קטנות בחלקו העליון של התמונה. דבר זה הינו טרייד אוף, בכךי קיבל טסיות המסומנות במלואן נctrack להקטין את הסף אך דבר זה יוסיף עוד רعش לתמונה וכאשר נגדיל את הסף להיות גדול יותר, לא יופיע הרבה רعش אך הטסיות לא יהיו מסומנות במלואן.

תשובה לשאלת 1.4 :

ニיצור מבנה בצורת דיסק על מנת לסנן את הטסיות מההתמונה ולקבל את הרקע:

Background

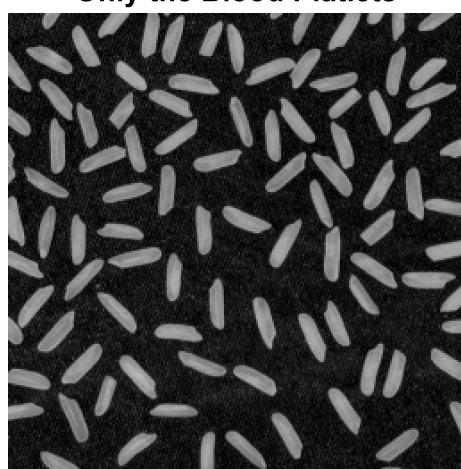


איור 4 : הרקע המתתקבל לאחר סינון הטסיות

באיור 4 ניתן לראות את הרקע המתתקבל ללא הטסיות. את רקע זה קיבלנו על ידי erosion עם מבנה בצורת דיסק ברדיוס של 10. הטסיות הקרובות להיות בצורה אליפטית ועל ידי מדידה ידנית של התמונה המקורית קיבלנו כי רדיוס המשנה (הקטן) שלחן הינו 5. בלקיחת מנגנון בעל רדיוס גדול יותר פי 2, אנו מבטיחים שהמבנה לא יתאים לשום חלק בטסית, ככלומר פועלות ה-erosion-platelets תעלים את הטסיות בצורה טובה.

נפחית את הרקע מתמונה המקור:

Only the Blood Platelets

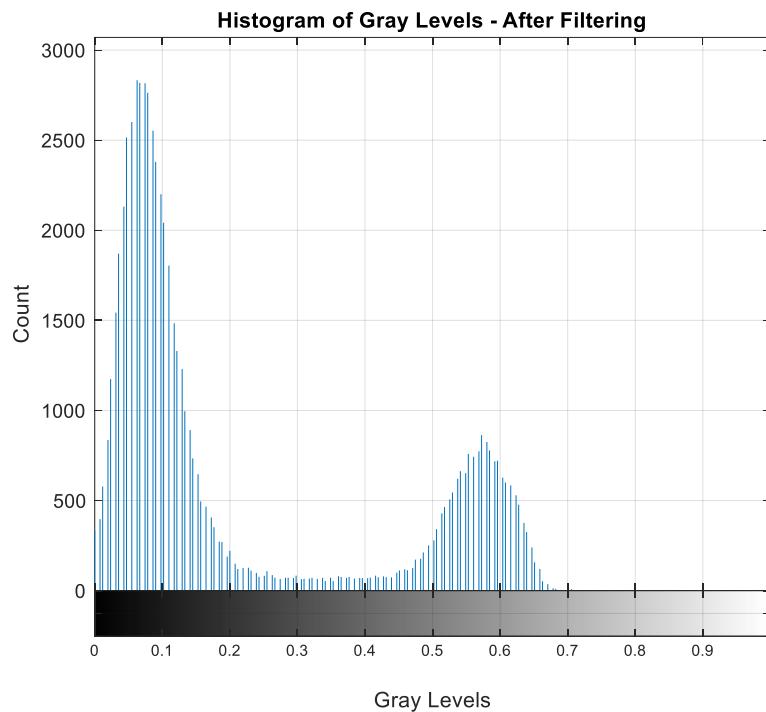


איור 5 : התמונה המתתקבלת לאחר הפחתת הרקע

באיור 5 ניתן לראות את התמונה המתתקבלת לאחר הפחתת הרקע מההתמונה המקורית. בשונה מאיור 1, הרקע של איור זה אחיד יותר בגונו דבר שהוא להפרדה טובה יותר של טסיות הדם מהרקע. בנוסף, נשים לב כי גווני האפור בתמונה כהים יותר כתוצאה מהחסרת הרקע.

תשובה לשאלת 1.5 :

כעת נבצע את הפעולות הקודמות שביצעו על התמונה החדש (איור 5). נציג את ההיסטוגרמת התמונה :

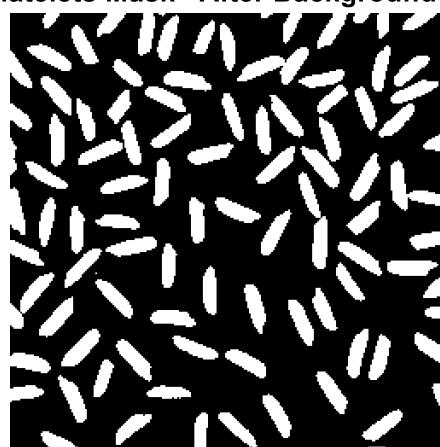


איור 6 : היסטוגרמת רמות האפור של התמונה החדשה שהתקבלה

באיור 6 ניתן לראות את ההיסטוגרמת רמות האפור. נשים לב כי בשונה מאיור 2, התפלגות הההיסטוגרמה כעת על טווח מצומצם יותר, בין 0 ל-0.7 בקירוב, וישנו 2 פיקים עיקריים המופרדים במרווח מובהק יותר. ככלומר, הפיק הימני (בסביבות 0.58) מייצג את טסיות הדם, בעוד שהפיק השמאלי (בסביבות 0.05) מייצג את רוב רקע התמונה.

כעת ליצירת המסכה בחרנו את הסף להיות 0.35 :

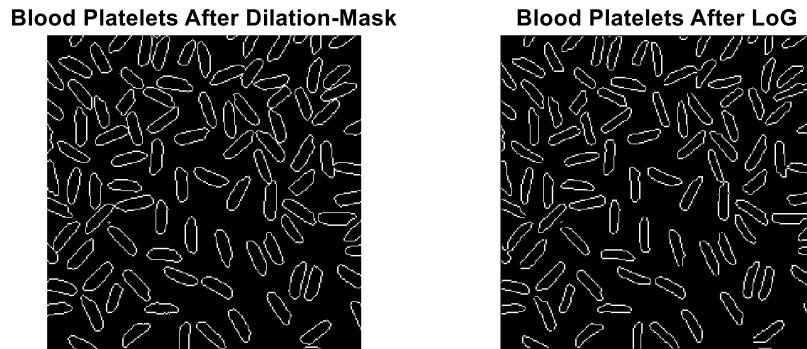
Blood Platelets Mask - After Background Removal



איור 7 : המסכה החדשה בעת בחרית הסף להיות 0.35

באיור 7 ניתן לראות את המסכה החדשה שיצרנו. בשונה מאיור 3, קיבלנו הפעם מסכה טובה יותר, ללא רעש ו特斯יות דם מסוימות במלואן. לא התחרשה בפערת זו טרייד אוף, لكن קיבלנו מסכה טובה יותר.

כעת נבעצע זיהוי גבולות בשתי דרכי ; החסרת התמונה לאחר dilation מהתמונה המקורית ובעזרת אופרטור LoG (Laplacian of Gaussian) נציג את שתי השיטות אחת ליד השניה :



אייר 8 : זיהוי גבולות הטסיות בשיטת LoG (ימין) וביצוע dilation וחסרתו מהתמונה המקורית

באייר 8 ניתן לראות את זיהוי גבולות הטסיות בשתי הדריכים הניל. ניתן לראות כי בשתי השיטות ניתן להזזה את טסיות הדם בצורה טובה.

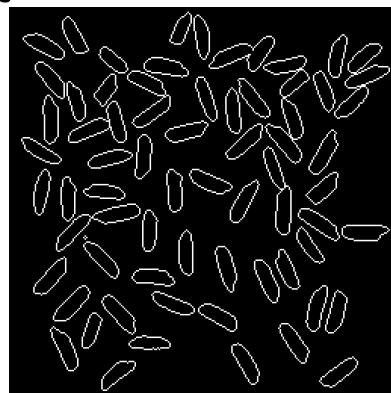
אוףן היישום של שיטת LoG מתבססת על 2 מסננים מרוחבים, מסנן לפלסייאן המדגיש שינויים בתדר גבוה (קצוות) אך רגיש לרעשים ומסנן גאוסיאני אשר מפחית רעשים וմבצע חילקה של התמונה. יישום מסננים אלו קורה בתחום התדר, כאשר נחשב את גרעין מסנן ה-G-GLoG באמצעות הכפלת גרעיני כל אחד מהמסננים הניל ובייצוע הפילטור על ידי כפל מטריצת המסנן בתמונה עצמה.

השיטה השנייה מתבססת על ביצוע dilation. בפועל זה ישנה הרחבה של האובייקטים בתמונה באמצעות מבנה מוגדר, כאשר הרחבת זו גורמת לטסיות לגודל ולמלא חלל גדול יותר סבירם. בעת החסרת התוצאה מתמונה המקורי, נקבל הדגשה של הפיקסלים הנמצאים רק באזוריים שהורחבו, ככלומר אלו שמהווים את גבולות הטסיות.

תשובה לשאלת 1.6 :

נסערץ את מספר הטסיות מתוך המסכה שייצרנו (אייר 7). תחיליה השתמשנו בפונקציית imclearborder על מנת להסיר מקומות התמונה את הטסיות שאין מופיעות בשלמותן. לאחר הסרת הטסיות התקבלה התמונה הבאה :

Clean Edge Detection - Number of Blood Plateles=60



אייר 9 : תמונה זיהוי הגבולות שהתקבלה לאחר שימוש בפונקציית imclearborder

באייר 9 ניתן לראות את תמונה זיהוי הגבולות לאחר שימוש בפונקציה הניל. נשים לב כי בשונה מאייר 8, הטסיות שנמצאו בגבולות התמונה הוחסרו והקוויים המתארת כל טסית הושלמו במלואן.

לאחר מכן, השתמשנו ב-bwlabel לקבלת מספר האלמנטים המצוים בתמונה. התקבל כי מספר הטסיות הינו 60.

בשלב השני של הניסוי, ביצעו מיצוי מאפיינים של טסיות הדם.

תשובה לשאלה 1.7 :

על מנת למצוא את הטסיות הנוטות ימינה בלבד, השתמש בפונקציה `regionprops` ובעזרתה נחלק את מאפייני התמונה ובניהם אוריינטציה, תומנת הטסיות לאחר مليוי והמסגרת הקטנה ביותר שמכילה את האובייקט. נסנן את הטסיות לפי האוריינטציה כאשר נבחר את הזווית להיות גדולה או שווה ל-0.

יצרנו מטריצה אפסים שתשתמש רקע לאכלס בתוכה את תומנות הטסיות הנוטות ימינה. בעזרת המאפיין `boundingbox`, נמצא את אינדקסי הפינה השמאלית העליונה של כל טסית ונאכלס את תומנות הטסית המלאה, שהתקבלה בעזרת המאפיין `filledimage`, בתמונה הרקע שייצרנו על ידי הוספה במקומות הרצוי.

Only Right Oriented Blood Plateles - Mask



איור 10 : המסכה שהתקבלה לאחר ביצוע השלבים הנ"ל

באיור 10 ניתן לראות את המסכה שייצרנו עבור הטסיות הנוטות ימינה. נשים לב כי רוב המוחלט של הטסיות אכן נוטה ימינה חוץ מאת אחת אשר נוטה שמאליה. לא הצלחנו לבצע החסרה של טסית זו מכיוון שגבולה חופף עם גבול של טסית הנוטה ימינה. הפונקציה `nnz` זיהתה את האובייקט כחיבור של שני הטסיות ולכון לא הייתה ניתנת לבצע הפרדה ביניהם.

תשובה לשאלה 1.8 :

מציג את תוכאת הכפלת המסכה הנ"ל בתמונה המקורית :

Only Right Oriented Blood Plateles - After Masking



איור 11 : הטסיות הנוטות ימינה

באיור 11 ניתן לראות את תומנת הטסיות הנוטות ימינה בלבד. תוכאה זו התקבלה לאחר הכפלת המסכה (איור 10) והתמונה המקורית (איור 1). בכל מקום במסכה לבן (ערך הפיקסל הוא 1) הפיקסל בתמונה המקורית עבר במלואו בעוד ששאר הפיקסלים (שחור ערך 0) יתראנסו וייהיו שחורים.

3.1.4 מסקנות

בניסוי זה התנסינו בסינון תמונה של טסיות דם עם רקע משתנה ושימוש במאפיינים למצירת מסיקה, במטרה ליצור תמונה המורכבת אך ורק מ特斯יות הנוטות ימינה (/). ניסוי זה בוצע בשני שלבים, סגמנטציה ומיוצי מאפיינים של טסיות הדם בתמונה.

בשלב הראשון של הניסוי, ביצעו סגמנטציה של טסיות הדם. בתחילת, יצירת המסכה התבכעה בעורת יצירתי היסטוגרמה עבור התמונה המקורית ובחרית סך מריד בין רמות האפור של הרקע לטסיות. לאחר מכן חורנו על שלבים אלו עבור התמונה המקורית פחותה בהשוואה לתמונה ללא הרקע. ראיינו כי המסכה שהתקבלה עבור התמונה המקורית אינכוטית פחותה בהשוואה ללא הרקע. מכז הסקנו כי סינון הרקע משפר משמעותית את אינכוטה המסכה. לבחירת מסכה נכונה ישנה חשיבות לביצוע סינון יעיל ולכן נעדר לחשטמש בשיטת החסרה לקבלת מסכה אינכוטית יותר. לאחר מכן, השתמשנו במסכה שיצרנו בשיטת החסרה ליזיהו גבולות הטסיות בשתי דרכים שונות; שימוש ב-G-Lo ושימוש ב-dilation של המסכה והחסרתה מהמסכה המקורית. ראיינו כי שתי השיטות הניבו תוצאה טובה (איור 8). לבסוף, נערכנו בפונקציה bwlabel לحصر הטסיות שאינן מופיעות בשלמותן מגבולות התמונה ובעזרת imclearborder התקבל כי מספר הטסיות המשוערכות בתמונה הוא 60.

בשלב השני של הניסוי, נערכנו בחילוץ מאפיינים בעורת regionprops למציאת הטסיות הנוטות ימינה בלבד, כאשר ביצעו את הפרדה על בסיס אוריננטציה. ראיינו כי התוצאות שהתקבלו אכן הניבו תוצאות טובות, כאשר התמונה הסופית שהתקבלה הכילה את כל הטסיות הפונוט ימינה וסיננה את הטסיות הפונוט שמאליה, למעט טסית אחת. הופעתה של טסית בודדת זו הפונה שמאליה התרחשה בשל חיפוי שטחה עם טסית הפונה ימינה הצמודה אליה ויזיהו שלhn כאובייקט יחיד. אנו סבורים כי秣ז זה התרחש כתוצאה מפעולות dilatation שבייצנו בעת יצירת המסכה. שיטת dilatation-mosion מtabבשת על הרחבת של האובייקטים בתמונה באמצעות מבנה מוגדר, יתכן כי בשל ההרחבה גבולות הטסיות השתנו באופן שהשפיע על הפרדת הטסיות. מכז למדנו על הצורך בשיפור שיטות היזיהו והסגמנטציה על מנת למזער טעויות ולשפר את דיוק התוצאות.

3.2 ניסוי 2

3.2.1 היפתוזה

בניסוי זה נבצע סינון של רעש 'salt&pepper' על ידי שימוש במסנן חציון בגדלים שונים ונבחן את השפעת גודל המסנן על סינון הרעש מההתמונה. נשרר כי קיים גודל מסוון חציון אופטימלי עבור התמונה המורעשת אותה אנו מסננים, המביא למינימום את שגיאת ה-MSE. נצפה כי מסננים קטנים מדי לא יסנו את הרעש בצורה יעילה וכי מסננים גדולים יטשטשו את התמונה יותר מדי, זאת מכיוון שמסנן חציון מחליף כל פיקסל בערך החציון של הפיקסלים החופפים (בהתאם לגודל המסנן) בין התמונה למסנן. כמו כן, נצפה לראות הבדל ברור בערך המובהקות הסטטיסטית לאחר חזרה על הניסוי מספר רב של פעמים.

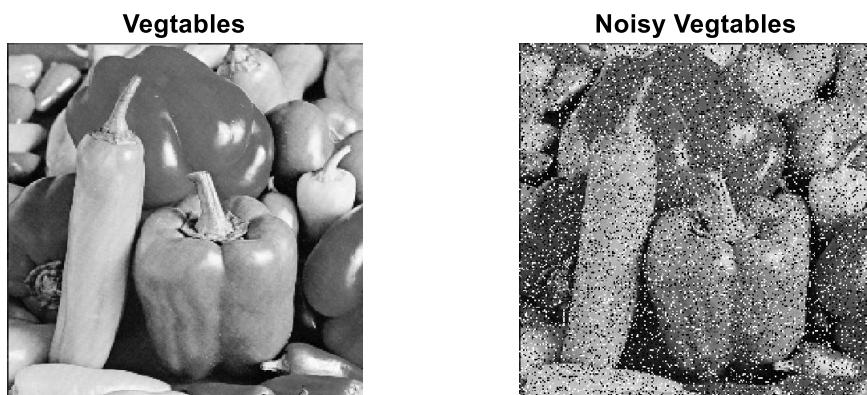
3.2.2 מתודולוגיה

כלים וחומרים : תוכנת המטלב.

מהלך הניסוי : בניסוי זה השתמש בתמונות המכשורות באתר הקורס, תМОנות ירקות ותמונה בה תיבלו את הירקות במעט 'salt&pepper'. השתמש בפונקציה שבינו בדוח המכון על מנת לשפר את התמונה, כאשר השתמש במסנן חציון בעל ממדים שונים (7×7 5×5 4×4 3×4 3×3 2×2 1×3 1×2) ועבור כל מסנן נחשב את ערך MSE בין התמונה המקורית לתמונה לאחר הסינון. נבחן את ערכי השגיאה המתקבלים ונבחר מהו הגודל האופטימלי עבור מסנן זה. נזכור על הניסוי 100 פעמים, נציג ממוצע וסטטיסטית תקן של השגיאה (MSE). נבחר מבחן סטטיסטי על מנת לבחון את ההבדלים בין המサンנים הריבועיים.

3.2.3 תוצאות

נציג את תМОונות הירקות עם הרעש וללא :

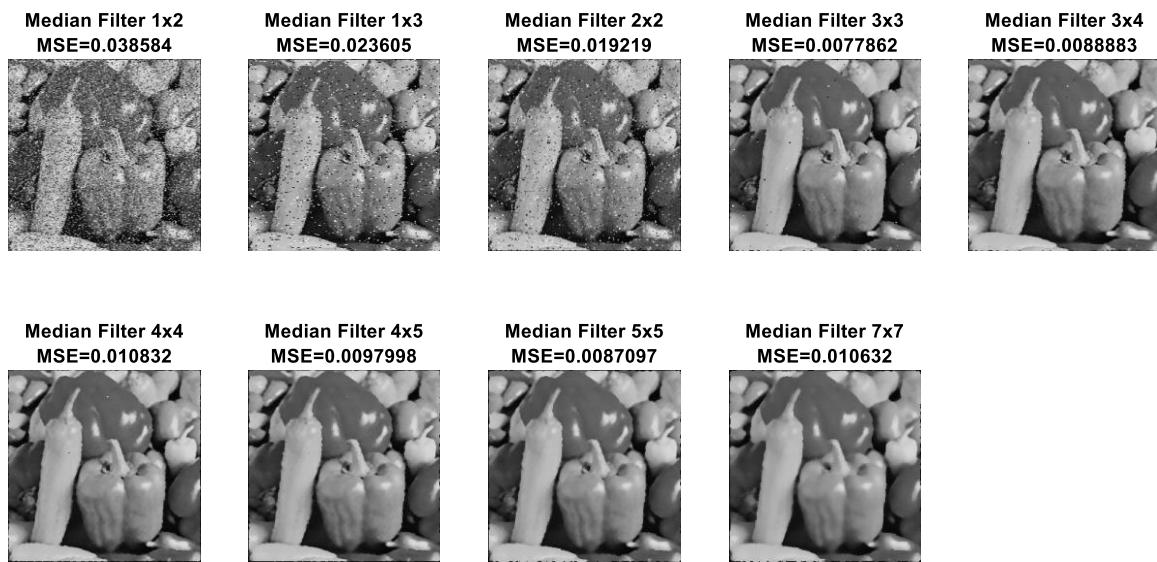


איור 12 : תМОונת הירקות הנקיה (שמאל) ועם רעש המלח פלפל (ימין)

באיור 12 ניתן לראות את תМОונת הירקות בשני מצבים ; נקייה מרוש (מקורית) ותמונה מורעשת (רעש מלח פלפל).

תשובה לשאלת 2.1 :

נשתמש בפונקציה שכתבנו בדוח המכין (CleanSP) על מנת לסן את הרעש ולשפר את התמונה השמאלית בצורה המיטבית. השתמשנו במסנן חציון בגודלים שונים. נציג את התוצאות :



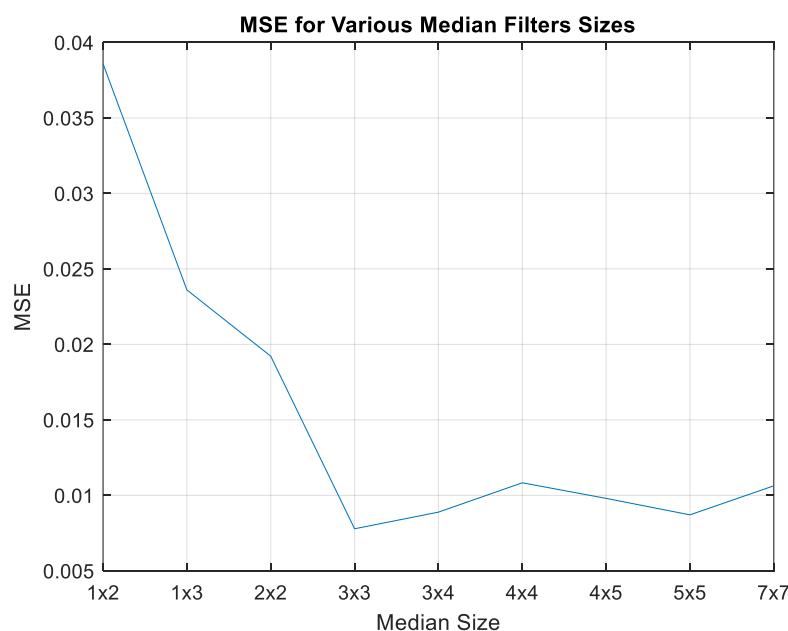
אייר 13 : תוצאות הסינון בעזרת מסנן חציון בגודלים שונים

באир 13 ניתן לראות את תוצאות הסינון בגודלי מסנן חציון שונים. נשים לב כי עבור מסננים בגודל קטן, הרעש לא סונן בצורה מיטבית ונראה לעין. ככל שהגודל המסנן גדול, כך ניתן להבחין פחות ברעש, אולם בגודלי מסנן גדולים מאוד נזכה בטשטוש יתר של התמונה.

לאחר מכן, חישבנו את ערך ה-MSE בין התמונה המקורי לתמונה אחרי הסינון [1].

$$(1) \quad MSE = \frac{1}{n} \sum_i (y_i - \hat{y}_i)^2 , \quad 0 \leq i \leq n - 1$$

להלן התוצאות המתקבלות :



אייר 14 : ערכי MSE המתקבלים עבור מסננים בגודלים שונים

באיור 14 ניתן לראות את ערכי ה-MSE שהתקבלו בשימוש במסנן חציו בגודלים שונים. ניתן לראות כי השגיאה המינימלית מתקבלת עבור מסנן בגודל 3x3, כאשר ניתן להבחין כי גוף ה-MSE הינו במוגמת ירידה עד לנקודה זו ולאחר מכן ישנה עלייה בערכי ה-MSE המתקבלים, אולם לא באופן ממשועוטי. ככלומר לפि גודל השגיאה המינימלי ולפי הסתכלות על מוצא המנסנים (איור 13), כך שモץ המסנן אינו מטושטש למדי ועוצמת הרעש נמוכה מאוד, גודל המסנן האופטימלי לפי קритריון MSE הינו 3x3.

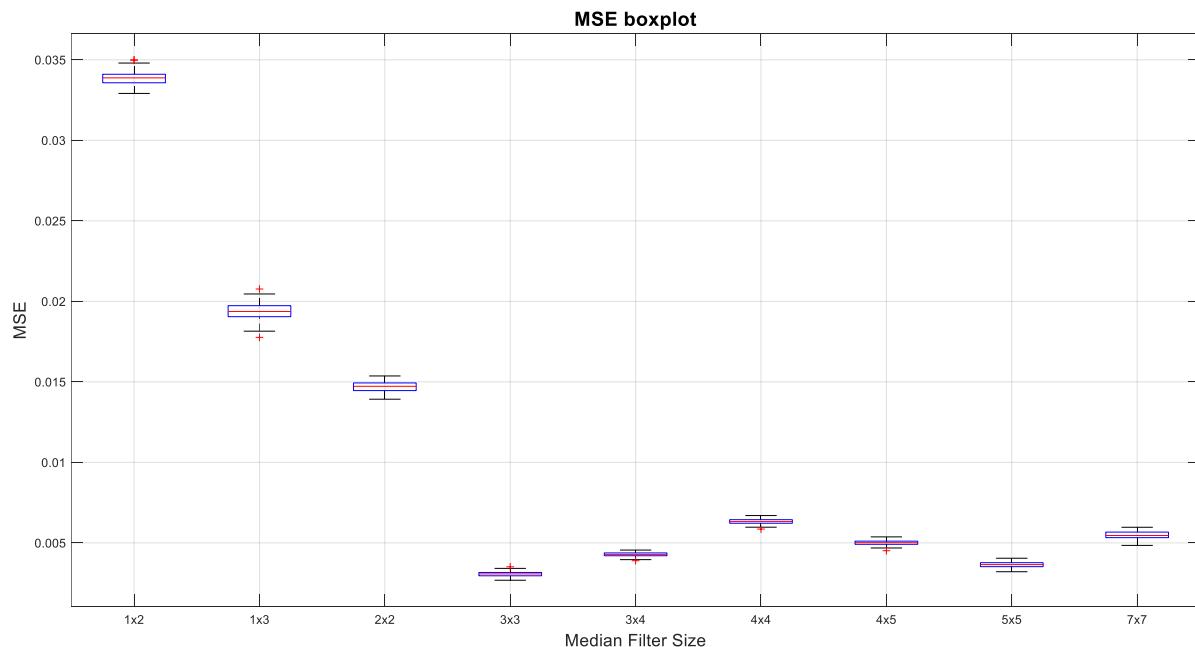
תשובה לשאלת 2.2 :

בעזרת הפונקציה `imnoise` יצרנו רעש מלאכותי אקריא של רעש מלח פלפל כאשר רמת הרעש היא 20% וחזרנו על החלק הקודם של הסיכון 100 פעמים. נציג את הממוצע וסטיית התקן :

טבלה 1 : ערכי ממוצע וסטיית התקן של ערכי ה-MSE

MSE	סטיית התקן	MSE	ממוצע	גודל מסנן
4.01×10^{-4}	0.0338			1x2
5.06×10^{-4}	0.0193			1x3
2.89×10^{-4}	0.0147			2x2
1.41×10^{-4}	0.0030			3x3
1.17×10^{-4}	0.0043			3x4
1.49×10^{-4}	0.0064			4x4
1.36×10^{-4}	0.0050			4x5
1.63×10^{-4}	0.0037			5x5
2.20×10^{-4}	0.0055			7x7

בטבלה 1 ניתן לראות את ערכי הממוצע וסטיית התקן של ערכי ה-MSE לאחר חוזה על הניסוי 100 פעמים.



איור 15 : גוף boxplot של ערכי ממוצע וסטיית התקן של ערכי ה-MSE עבור גדי המסנן השונים

באיור 15 ניתן לראות את גוף Boxplot לאחר החזותה הרעש וסינונו 100 פעמים. מהתבוננות בגוף זה ובטבלה 1 ניתן להבחין באותה מוגמת של איור 14, ככלומר ביצוע החזרות לא השפיע על סדר טוב הסיכון לפי קритריון MSE. סטיית התקן (המודיפה בגוף כתחום פסים שחורים) עבור מסננים בגודל קטן (עד גודל 2X2) גדולה יחסית בעוד שסטיית התקן של המסננים בגודל גדול יותר קטנה. אין חפיפה משמעותית בין תחומי סטיית התקן כלומר, נוכל להבדיל סטייטיסטי בין המסננים בגודלים השונים. נשים לב כי ישנו מספר outliers, המשורנים בפלוס אדום, אך הם בודדים וקרובים מאוד לטוחה סטיית התקן, ככלומר אינם ישפיעו בצורה רבה על תוצאות הניסוי.

ביצענו מבחן סטטיסטי מסוג one-way ANOVA על מנת לעמוד על ההבדלים בין המサンנים הריבועיים. קבענו את השערת האפס להיות כך שאין הבדלים סטטיסטיים משמעותיים בין סינון המサンנים בגדים הריבועיים השונים. נציג את תוצאות המבחן זה:

```
{'Source' }      {'ss' }      {'df' }      {'MS' }      {'F' }      {'Prob>F' }
{'Columns'}     {[ 0.0087]}    {[ 4]}     {[ 0.0022]}    {[5.4416e+04]}    {[ 0]}
{'Error' }      {[1.9892e-05]}  {[495]}    {[4.0185e-08]}  {0x0 double }  {0x0 double}
{'Total' }       {[ 0.0088]}   {[499]}    {0x0 double }  {0x0 double }  {0x0 double}

p-Value=0
```

איור 16 : תוצאות מבחן *p-Value*-ו one-way ANOVA

באир 16 ניתן לראות את תוצאות המבחן הסטטיסטי. מהטבלה ניתן לראות כי המבחן מראות כי יש הבדל מובהק בין הקבוצות השונות. ערך ה-*F*-statistic המתקבל גדול מאוד, מסדר גודל של 10^4 , וערך *p-Value*-ו קטן מ-0.05 (שגיאת α).

3.2.4 מסקנות

בניסוי זה בחנו את השפעת גודל גרעין מסווג חציון על סינון רعش מלח פלפל מתוך תמונה. לצורך כך השתמשנו בפונקציה CleanSP ליצירת מסנני חציון בגדלים שונים, הפעילו את המサンנים השונים לשינון הרעש ובחנו את השפעתם על התמונה לאחר סינון, תוך חישוב שגיאת ה-MSE בין התמונה המקורית לתמונה לאחר סינון. ראיינו כי באופן ויזואלי, ככל שהשתמשנו במסנן גדול יותר, כך הרעש פחות והתמונה הפכה למושחתה יותר. מהතוצאות שהתקבלו (אייר 14) הסקנו כי במקרה זה המסנן בגודל 3x3 הניב את התוצאה הטובה ביותר, ככלומר את היחס הטוב ביותר בין סינון הרעש לשמירה על חזרות התמונה. בביבטוע 100 חזרות על הניסוי ראיינו כי נפתחה עקבות בתוצאות, כאשר ממוצע התוצאות שהתקבל היה קרוב לערך ה-MSE של הניסוי המקורי, עם פיזור קטן ושמירה על מגמת הגרפ. בנוסף, ביצענו מבחן ANOVA סטטיסטי של MSE של מסננים חציוניים ריבועיים [2]. מבחן זה הראה כי קיים הבדל מובהק להשוואת ערכי MSE של מסננים חציוניים ריבועיים. מהතוצאות בין תוצאות סינון רعش מלח פלפל על ידי מסננים חציוניים ריבועיים בגדלים שונים. מהතוצאות שהתקבלו ניתן להסיק כי שימוש במסנן חציון בגודל 3x3 הינו האופטימלי לשינון רعش מלח פלפל מהתמונה הנתונה. אולם, לא נוכל להסיק כי מסנן זה הינו האופטימלי לכל המקרים. ניסוי זה בוחן רק תמונה אחת ורמת רעש אחת, ככלומר ייתכן ויתקבלו תוצאות שונות עבור תונות שונות ורמת רעש אחרות. כמו כן, בוחנו שימוש במסנן חציון ללא השוואה למסננים אחרים לשינון הרעש.

ניסוי 3 3.3

היפותזה 3.3.1

בתרגיל זה כתבנו מספר פונקציות לצורך ניתוח סרטון אשר משתמש בהם על מנת ליזות את מיקומי העיניים המבוקשות במספר מקומות שונים. בניסוי זה ננתה את הסרטון הויידאו שצילמנו של תנועת האישון, כאשר נרצה לגלוות את המספר עליו מסתכלות העיניים באותו פריים. המטרה הייתה לכתוב אלגוריתם, שיזהה על סמך בסיס תנועות כיוול, לאיזה כיוון האדם הסרטון מסתכל. נצהה כי לא נצליח לקבל זיהוי מושלם של כל הפרויימרים הסרטון, הנובעת מחוסר אידיאליות בעת ביצוע ניסויי אמפירי מסווג זה, כגון שינוי במיקום ראש הנבדק בין תנועות הכיוול הסרטון, מצמצמים ושגיאות הן ביצוע הכלול והן ביצועי הנבדק, ראש חיצוני שמשפיע על התמונה וכיוול על בסיס מספר תMOVNOT קטן יחסית.

3.3.2 מתודולוגיה

כלים וחומרים: סרטון וידאו ותמונות, תוכנת המطلب.

מהלך הניסוי: ראשית ווידאו שהפלורנסטים בחדר כבויים על מנת למנוע רעש-ב-[Hz] 50. לאחר מכן קיבענו את צווארו של הנבדק לשולחן הניסוי ומיקמו את המצלמה מול עיניו. פתחנו את כובץ המטלב ומקסמו את איקות התמונה בעזרת כיוונו פרטורי המצלמה דרך GUI. לאחר מכן צילמנו 4 תמונותCiyo אשר בכל אחת מהן העיניים מביאות אל מספר אחר מ-1 עד 4, חזרנו על פעולה זו 12 פעמים. בתמונות אלו נשתמש ליצירת מטריצת Ciyo למיקומי העיניים. לבסוף, צילמנו וידאו של 30 שניות כאשר בכל 2 שניות מיקום העין משתנה לפי הסדר הבא: 2→1→2→3→2→1→2→3→4→2→3→2→3→1.

3.3.3 תוצאות

תשובה לשאלת 3.1:

נציג את הפונקציה שכתבנו המחזירה את מיקום כל עין בפריים שנשלח אליה:

```

function [Eye_Pos]=EyePosition_FUNC(Frame)
% [Eye_Pos]=EyePosition_FUNC(Frame)
%
% This function returns the eyes position from the input frame
%
% Inputs:
% Frame      - Desired frame for extraction eye position. Matrix of
%               normalized gray scale image
%
% Outputs:
% Eye_Pos    - The eye position of both of the eyes. Returns as a matrix 2x2
%               where the first row is the Left eye and second row is the Right
%               eye. in each row there is the coordinates (x and y indexes of
%               the image)

% Assining print variable. Change to 1 for plotting image
print=0;

% Filtering the desired frame with median filter of size 45x45
adjFrame=CleanSP(Frame, 'Median', 45, 45);
% Printing image
if print
    figure('WindowState', 'maximized')
    nexttile
    imshow(adjFrame)
    title('Filtered Image for Detection Eye Position')
end

% Finding circles
min_rad=20;
max_rad=30;

```

```

centers=imfindcircles(adjFrame,[min_rad max_rad], 'ObjectPolarity','dark' ...
    , 'EdgeThreshold',0.01, 'Method', 'TwoStage');
% Rounding for indices
Eye_Pos=round(centers);

% Sorting the circle indices according to their gray scale value
[~,ind]=sort(sum(adjFrame(Eye_Pos(:,2),Eye_Pos(:,1)).*eye(length(Eye_Pos))), 'ascend');
% Extracting only the blackest indices
Eye_Pos=Eye_Pos(ind(1:2),:);

% Sorting the rows as left in first row and right in second
[~,I]=sort(Eye_Pos, 'descend');
Eye_Pos=Eye_Pos(I(:,1),:);
end

```

מציג תוצאה אחת של מיקומי העיניים :

Eye Detection Example



איור 17 : דוגמת תוצאת הפעkcיה עבור מיקום העיניים

באייר 17 ניתן לראות דוגמא אחת למצאה הפעkcיה עבור זיהוי מיקום העיניים. נשים לב כי התמונה זיהתה את מיקום האישונים בצד שמאל, כאשר הוכבהת האדומה נמצאת על מרכז האישון והעיגול הכהה הוא על מנת להראות את כל מיקום העין. חתכנו את התמונה לפני הכנסתה לפעkcיה על מנת למקד את הפעkcיה רק לאזור העיניים ולהפחית את כל רעש הרקע למנוע זיהוי שגוי.

תשובה לשאלה 3.2 :

הפעkcיה הנ"ל מוציאה את מיקום כל עין בפריים שנשלח אליה בצד שמאל :



איור 18 : דיאגרמת בלוקים של אלגוריתם הפעkcיה EyePosition_FUNC

באיור 18 ניתן לראות את דיאגרמת הבלוקים של האלגוריתם של הפונקציה הנויל. תחילת אנו מסננים את התמונה במסנן חציון בגודל 45x45. סינוון זה מבצע טשטוש של התמונה ויסנן רעשים. דבר זה יגביר את זיהוי המיקום של האישון מכיוון שמיוקום העיניים שיתקבל יהיה עיגולים בולטים יותר וכחים יותר מאשר חלקה התמונה. לאחר מכן נקבע את רדיוס העיניים להיות בין 20 ל-30 פיקסלים. את הממדים אלו מדנו בצורה ידנית על ידי פונקציית `imdistline`. לאחר מכן, השתמשנו בפונקציית `imfindcircles` על מנת למצוא את כל העיגולים הכהים בתמונה. מוצא הפונקציה הינו המיקום בתמונה המקורית ולכן על מנת להפוך אותו לאינדקסים, נבצע עיגול כלפי מעלה [3]. קיבלו מספר רב יחסית של אינדקסים המבטאים עיגולים כהים בתמונה, וכן על מנת למצוא אך ורק את מיקומי העיניים, נמצא את המיקומים שהתקבו בהם גווני האפור הינם הכהים ביותר. נחלץ את שני העריכים הכהים ביותר ונסדר את מטריצת המוצא להיות כך שתתאים לערך המצופה. השורה הראונה (עין שמאל של הנבדק) תהיה בעלת ערך ה-X הקטן יותר. הגובה יותר והשורה השניה (עין ימין של הנבדק) תהיה בעלת ערך ה-X הגדולה.

תשובה לשאלה 3.3 :

כתבנו את הפונקציה המחזירה מטריצת כיוול עבור מערכת התאים שנשלחה אליה. קלט המערכת שיצרנו הינו מערך תאים בגודל 1×4 אשר כל תא מייצג את המספר עליו הנבדק הסתכל (1,2,3,4). בכל תא נמצא מערך נוסף של תאים בגודל 12×12 כאשר בכל תא ישנה מטריצה המייצגת כל פריים חתוך של מיקום העיניים המסתכל על הערך המספרי בהתאם. נציג את הפונקציה שכתבנו :

```

function [EyeCalib] = EyeCalibration_FUNC(CalibFrame)
% [EyeCalib] = EyeCalibration_FUNC(CalibFrame)
%
% This function returns the eyes position calibration matrix from the input
%
% Inputs:
% CalibFrame      - Cell array 4x1 in which each cell is 1x12 cell array of
%                   cropped frames
%
% Outputs:
% EyeCalib        - The eye position calibration matrix of both of the eyes.
% Returns as a matrix (N-1)x8x2 where N is the number of
% calibration frames, 8 is the number of coordinates in each
% position (2 [coordinates]*4 [positions]) and 2 is the
% dimantion of the eyes (1 for each eye for calibration).

% Assining print variable. Change to 1 for plotting image
print=0;
% Assigning variables and Preallocating memory
N=length(CalibFrame{1});
left=1;
right=2;
EyeCalib=nan(N-1,8,2);
for i=1:4
    for j=1:N-1
        % Extracting frames
        Frame=CalibFrame{i}{j};
        % Finding eye position through our position function
        Eye_Pos=EyePosition_FUNC(Frame);
        % Allocating the eye position to our output
        EyeCalib(j,[2*i-1 2*i],left)=Eye_Pos(left,:);
        EyeCalib(j,[2*i-1 2*i],right)=Eye_Pos(right,:);
        % Printing results
        if print
            nexttile
            imshow(Frame)
            hold on
            viscircles(Eye_Pos,28,'EdgeColor','b');
            plot(Eye_Pos(1,1),Eye_Pos(1,2),'r*')
        end
    end
end

```

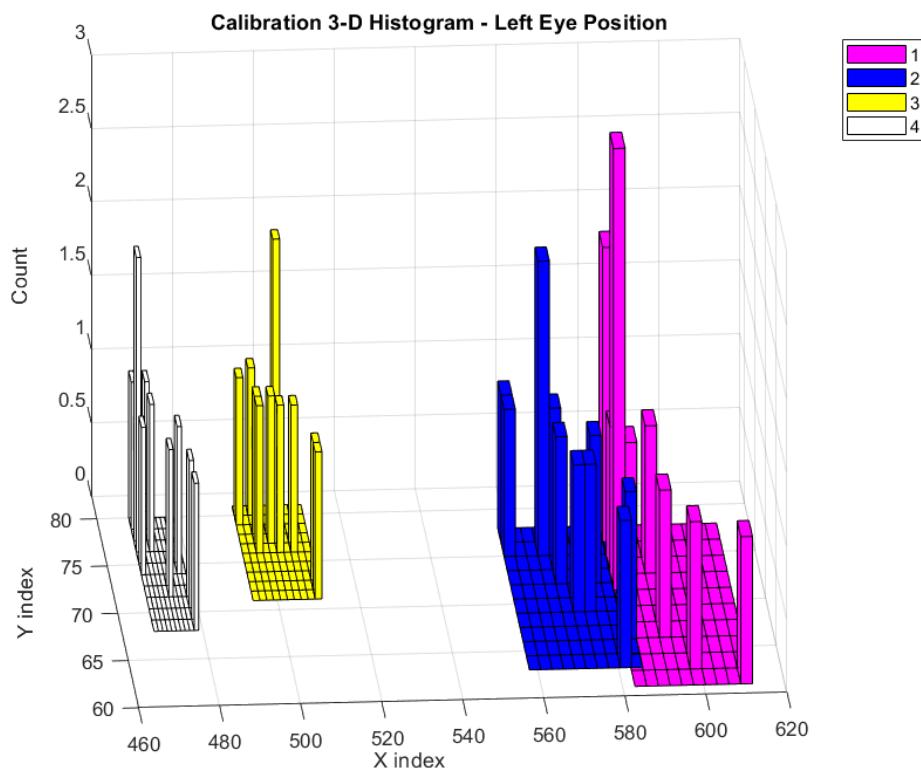
```

plot(Eye_Pos(2,1),Eye_Pos(2,2), 'r*')
end
end
end

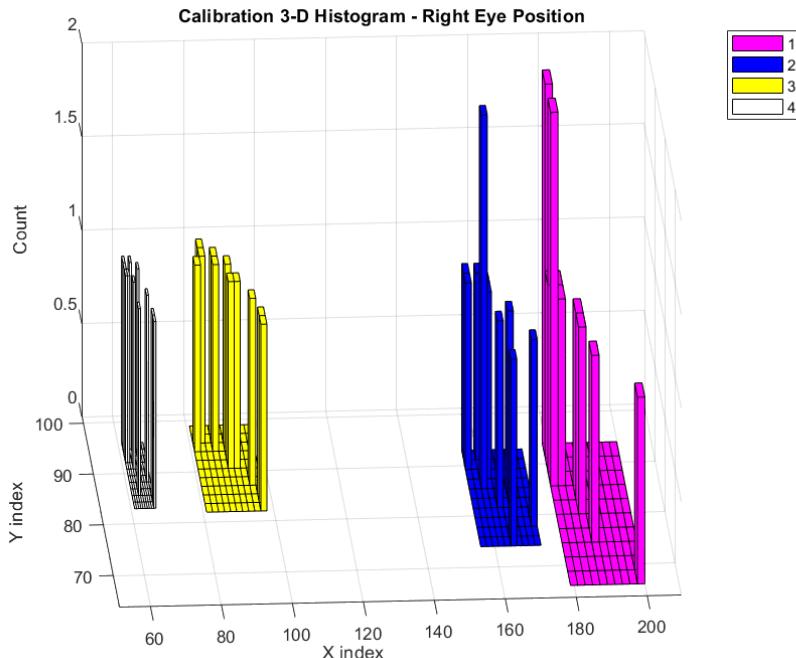
```

פונקציה זו מקבלת את מערך התאים כפי שהסבירנו לעיל ומחזיא את מטריצת הcyol כמטריצה של תלת ממדית. המימד השלישי הינו מימד מסדר 2 כאשר בכל אחד מהן תהיה מטריצת cyol לעין אחת. המטריצת cyol תסדר את אינדקסים מיקומי העיניים, שהתקבלו בתוצאה מהפעלת הפונקציה הקודמת, בהתאם למספר עליו הנבדק הסタル. בנוסף, נבחן רק N-1 תМОנות פריאמיים (במקרה שלנו 11 מתוך 12 תМОנות פריאמיים) עבור כל הסタルות של הנבדק על מספר ספציפי ליישום מטריצת הcyol.

נציג היסטוגרמות של מטריצות הcyol עבור 2 העיניים :



איור 19 : היסטוגרמת cyol עין שמאל



אייר 20 : היסטוגרם כיוון עין ימין

באייר 19 וbaior 20 ניתן לראות את ההיסטוגרמות מטריצות הכיוול עבור 2 העיניים, שמאל וימין בהתאם. תחילת נשים לב כי ישנה הפרדה רחבה מאוד בין מיקומי 1 ו-2 לבין 3 ו-4- במיוקמי אינדקס ציר ה-X בעוד שמיוקומי ציר ה-Y בעין ימין ועין שמאל אינם מופרדים כמעט כלל ביכול פיזי של מיקום העיניים. כמו כן, נבחין כי בין מיקומי 3 ו-4 ישנה הפרדה מסוימת לחולוטין באינדקס ציר ה-X בשתי העיניים בעוד שבין מיקומי 1 ו-2 הה הפרדה היא קטנה ואיפלו חופפת במקצת במרקחה של עין שמאל. יתר על כן, נשים לב כי בכל מקבץ של מיקום, התפלגות מיקום העין רחבה בציר ה-Y בעוד שבציר ה-X הינה צרה. בנוסף, בכל מקבץ ישנו פיק גדול מ-1 המסלל כי העין חוזרת בערך לאותו מיקום בציר ה-X וה-Y.

תשובה לשאלה 3.4 :

נציג את הפונקציה שכתבנו מקבלת פריים בווד מטריצת כיוול ומחזירה את המספר עליו מסתכלות העיניים :

```

function [Eye_Look] = EyeLook_FUNC(Frame,EyeCalib)
% [Eye_Look] = EyeLook_FUNC(Frame,EyeCalib)
%
% This function returns the eyes position number from a single frame and a
% specific calibration matrix
%
% Inputs:
% Frame          - Desired frame for extraction eye position. Matrix of
%                  normalized gray scale image
% EyeCalib       - The eye position calibration matrix of both of the eyes.
%                  Matrix of (N-1)x8x2 where N is the number of calibration
%                  frames, 8 is the number of coordinates in each position
%                  (2 [coordinates]*4 [positions]) and 2 is the dimantion
%                  of the eyes (1 for each eye for calibration).
%
% Outputs:
% Eye_Look      - A matrix of 1x2 housing the number the left and right eye
%                  has looked on
%
% Assining print variable. Change to 1 for plotting image

```

```

print=1;
% Left and right indices
Left=1;
Right=2;
% Eye position calculation
Eye_Pos=EyePosition_FUNC(Frame);

% Calculating the mean of each index from the calibration matrix
LeftMean=mean(EyeCalib(:,:,Left));
RightMean=mean(EyeCalib(:,:,Right));
% Preallocation memory and loop for calculating the distance between the
% current eye position from the mean position for each state (number)
[LeftError,RightError]=deal(zeros(1,4));
for i=1:4
    LeftError(i)=vecnorm(Eye_Pos(Left,:)-LeftMean([2*i-1 2*i]));
    RightError(i)=vecnorm(Eye_Pos(Right,:)-RightMean([2*i-1 2*i]));
end
% Finding the minimum distance index
[~,LeftInd]=min(LeftError);
[~,RightInd]=min(RightError);
% Returning the number each eye was looking on
Eye_Look=[LeftInd,RightInd];
end

```

תשובה לשאלת 3.4.1:

פונקציה זו מקבלת פרויים ספציפי ואת מטריצת הcoil שיצרנו סעיף קודם ומחזירה את המספר עליו כל עין הסתכלה. את המיקום נקלט באמצעות ניתוח סטטיסטי. הניתוח הסטטיסטי הנבחר מבוסס על חישוב מרחק אוקלידי בין ממוצעי הקוארדינטות (y, x) לבין מיקום העין שהתקבל בפרויים הנבחר. המרחק האוקלידי מחושב כך [4]:

$$(2) \quad d(p, q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2}$$

כאשר p ו- q הינו נקודות במרחב דו מימדי ($p = (p_1, p_2)$, $q = (q_1, q_2)$) $= c$ כאשר כל קורדינטה היא של ציר $-x$ וה- y בהתאם. בחירת הנקודה למדידת מרחק להיות p או q הינה שרירותית, שכן ה斛לה החישור בריבוע אינו משנה את סדר קביעת הפרמטרים.

בחירת ניתוח זה הייתה בשל פשוטותו ויעילותו. כאמור, חישובו פשוט ואינו דורש הנחות סטטיסטיות מיוחדות. ביחס ניתוח זה נוכל בדרך פשוטה לחוש את המספר בעל התכונות הגבוהה ביותר עליו הסתכל המשתמש, כיוון שככל שהמרחק האוקלידי בין מיקום העין שהתקבל לבין ממוצעי הקוארדינטות שהתקבלו בכיוול קטן יותר כך גודל הסיכוי שבחירת כיוון ההסתכלות יהיה מדויק.

תשובה לשאלת 3.4.2:

בחנו את הפונקציה שיצרנו באמצעות 4 פרויים שונים אשר כל אחד מסתכל על מספר שונה. הפרויים הללו לא שומשו ביצירת מטריצת הcoil. נציג את התוצאות המתקבלות עבור כל עין:

טבלה 2: השוואה בין זיהוי מיקום הסתכלות העיניים ולאזרו הסתכלות הידוע

אזור הסתכלות	عين يمين	عين شمال
1	1	2
2	2	2
3	3	3
4	4	4

בטבלה 2 ניתן לראות את תוצאות הפונקציה שהתקבלו מאربעת הפרויים הנ"ל. נשים לב כי עבור אזור הסתכלות 2,3 ו-4 הפונקציה מצאה את המספר הנכון עבור 2 העיניים בהצלחה. עבור אזור הסתכלות 1 קיבלנו הבדל בין עין شمال לעין ימין. עין شمال Ziיתה אזור שגוי אך סמוך בעוד שעין ימין Ziיתה את האזור בהצלחה.

תשובה לשאלת 3.6 :

חישבנו את אחוז השגיאות עבור כל עין :

טבלה 3 : אחוז שגיאות סיוג כל עין

שגיאה עין שמאל		שגיאה עין ימין	
29.32%		6.13%	

בטבלה 3 ניתן לראות את אחוז השגיאות עבור כל עין. נשים לב כי עבור עין ימין התקבל אחוז שגיאות גובה מאוד יחסית לעין ימין. על מנת לבחון את התוצאות הללו נציג את התוצאות בצורת מטריצת מבוכה :

Confusion Matrix for Left Eye - Error Percentage=29.327

		True Class			
		1	2	3	4
True Class	1	105	3		12
	2	39	74		70
True Class	3		2	113	7
	4		1	31	
		Predicted Class			
		1	2	3	4

Confusion Matrix for Right Eye - Error Percentage=6.126

		True Class			
		1	2	3	4
True Class	1	117	3		
	2	20	162	1	
True Class	3		2	119	1
	4			1	31
		Predicted Class			
		1	2	3	4

איור 21 : מטריצות המבוכה עבור 2 העיניים

באיור 21 ניתן לראות את מטריצות המבוכה אשר התקבלו עבור סיוג 2 העיניים. נשים לב כי עבור עין ימין (ימין) קיבלנו זיהוי גובה מאוד של true classes 1,3,4 class במיוחד ב- classes 1,3,4. עבור class 2 קיבלנו דיווק גבוה אך עם הרבה זיהויים שגויים. האלגוריתם זיהה את class 2 כ- class 1- class 1 כ- class 2. לעומת זאת עבור עין שמאל (שמאל) קיבלנו הרבה זיהויים שגויים. עבור class 2, האלגוריתם זיהה אותו הרבה כ- class 1. לעומת זאת זיהה גם כ- class 4 כ- class 3. בנוסחא 1 class 1 או class 4 מס' פעמים. דבר זה הגדיל לנו את השגיאה של עין זו במיוחד.

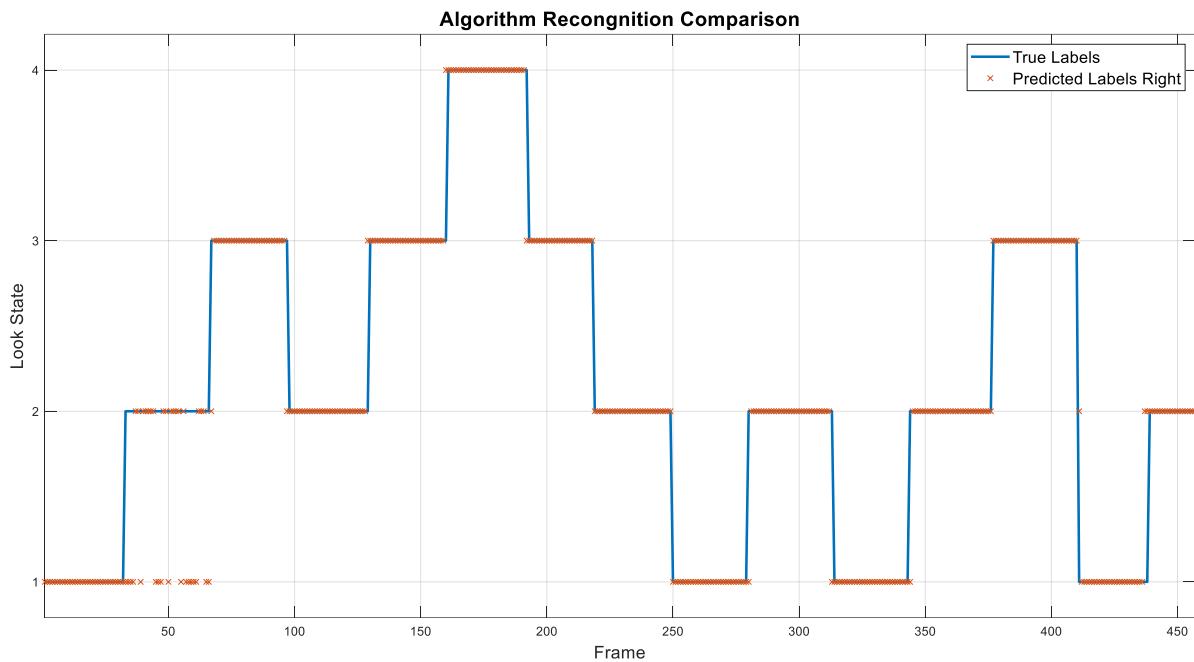
Failed Detection of Left Eye From a Video Frame - Class 2



איור 22 : דוגמא למציאת מיקום העיניים עבור פריים בודד הנמצא במיקום 2

באיור 22 ניתן לראות דוגמא כושלת למציאת מיקום עין שמאל באחד מהפריטים של class 2. נשים לב כי מיקום עין שמאל זיהה את מיקום המיקום הלא נכון, מיקום שהוא חלק מעין ימין. דבר זה גורם לכך שערך אינדקס ציר ה-x היה רחוק למדי (קטן מאוד) מערכיו מטריצת היכול עבור העין הנשייל (איור 19), ככלומר המרחק המינימלי ממוצע הקבוצה ביותר לעין ימין יוביל לכך שהפריטים מסווג זה יסועו כקבוצה 4.

נציג גראף המתאר את תוצאות האלגוריתם עבור עין ימין אל מול ה-labels true.



איור 23 : השוואה בין תוצאות האלגוריתם עבור עין ימין לערכיהם האמיטיים

באיור 23 נראה את השוואת תוצאות האלגוריתם (מייצגות כאייסס כתום) לבין הערכיהם האמיטיים (מייצגים כגרף צף כחול). נשים לב כי תוצאות האלגוריתם תואמות לערכיהם האמיטיים ברוב מהמרקם כאשר ישן 2 מקרים בהן יש שגיאות ; בעת מעברים בין מספר true-labels ובחלקו הראשון של הניסוי עלייה סבב פריים .50.

3.3.4 מסקנות

בניסוי זה ביצענו סיווג של כיווני הסטקלות שונים. לשם כך השתמשנו ב-11 תמונות כיוול עבור כל כיוון, מציאת מקום העיניים בכל כיוון ויצירת מטריצת כיוול לכל עין. לבסוף, השתמשנו במטריצת הכוול שבנו והתאמות ידניות לצורך סיווג כיווני הסטקלות של הנבדק בסרטון וידאו. בניתוח תמונות הכוול, בהצגת היסטוגרמות פרישת האזוריים לכל כיוון ראיינו כי קיימת הפרדה ברורה יותר בין כיווני הסטקלות עבור עין ימין (איור 20), لكن נסיק כי הסיווג בעזרתו עין ימין יהיה אינטוי יותר.

בחנו את טיב הפונקציה שיצרנו בעזרה מוצא הפריים שלא בכלל (הפריים ה-12 עבור כל עין). ראיינו כי עבור עין ימין הדיקוק שהתקבל היהנו 100%, לעומת זה היו ארבעת הכוונים באופן מדויק, ואילו עבור עין שמאל הדיקוק שהתקבל היהנו 75%, כאשר התקבלה שגיאת סיווג של אзор הסטקלות 1 כאזרור הסטקלות 2. הסקנו כי סביר ששגיאה זו נובעת בשל אזור החפיפה הקיים בין אזוריים סמוכים אלו עבור עין שמאל (איור 19).

בניתוח הסרטון קיבלנו אחוזי הצלחה עבור 2 העיניים ; עין ימין ו-10.68% עין שמאל. ניתן להסיק כי ההבדל באחוזי ההצלחה בין 2 העיניים נובע מ-2 סיביות ; קבוצות התפלגות אינדקסי מיקומי העיניים בתמונות הכוול היו מופרדות יותר לפי ציר ה-א בעין ימין מאשר בעין שמאל וזיהוי העיניים בעין שמאל עבור סיווג 4 class או לאו היה מדויק. על פי איור 22 ניתן לראות כי מיקום עין שמאל עבור פריים בודד מ-2 class סיווג כ-4 class עקב בשל בזיהוי מיקום עין שמאל. נסיק בעזרה איור 21 כי דבר זה קרה מספר רב של פעמים עקב כך שישנם הרבה סיווגים שגויים ל-4 class מכל שאר ה-classes (במיוחד מ-2 class). ניתן כי חוסר הדיקוק בזיהוי מיקום עין שמאל נובע ממשתנים כגון תאורה אחרת בשל מנת הפנים של הנבדק במהלך צילום הסרטון.

הסיבות הנ"ל פגעו רבות באחוז ההצלחה של זיהוי עין שמאל, אך עין ימין זוהתה באחוזי הצלחה גבוהים מאוד אולם לא מושלמים. ניתן להסיק כי ישן מספר סיבות אפשריות לשגיאות הסיווג הכלליות שהתקבלו ; ראשית, מיקום הנבדק בצילום הוידאו השתנה ביחס למיקומו בתמונות הכוול. ניסינו לתקן היסט זה באמצעות חיתוך התמונה ושינוי הגודל על מנת לקבל את מיקום הנבדק הזזה ביוטר למיקום הנבדק בתמונות הכוול ככל הניתן. אולם גם לאחר ביצוע התאמות בצורה ידנית, נסיק כי התיקון אינו

מדוק מספיק. כמו כן, הבחנו כי מרבית השגיאות הינו בחלוקת הראשון של הסרטון, בסיווג שגוי של אзор הסתכילות מס' 2 - השכיח מבין אзорיו הסתכילות בסרטון, כאזור הסתכילות מס' 1 - אזור הסמוך לאזור הסתכילות 2 (איור 23). נסיק כי באזור זה היו הפרעות הנובעות ממנה הנבדק כתזזה, מצטוץ או רعش חיצוני רב שפגעו ביכולת הזיהוי במקטע ספציפי זה. למרות זאת, קיבלנו אחוזי הצלחה גבוחים מאוד שכן ניתן להסיק כי נוכן להסתמך בעיקר על עין ימין לצורך זיהוי הסתכילות הנבדק.

בנוספ', בניגוד לציפויינו, כי נראה שגיאות בולטות לעין בעת מעברים בין ציוני הסתכילות שונים בשל טשטוש התמונה וקשיי לזיהוי, ראיינו כי ישנן שגיאות מועטות בחלוקתם אלו (איור 23).

3.4.1 היפוזה

בניסוי זה ננתה את תנועת רגלי ימין במהלך הליכה במהירות של 2 קמ"ש, תוך שימוש במרקרים עיגולים לבנים על רקע שחור על מנת להקל על ביצוע הניתות. נצפה כי נצליח לזהות את המרקרים אך לא באופן מדויק, זאת מכיוון שהפריים המנוטחים הינם מtoo סטטי, כלומר כאשר הנבדק בתנועה הפריים המתקבל עשוי להיות מטוושט ולמנוע זיהוי מיטבי של המרקרים. לכן בפריים המוטושטשים נצפה לראות יותר שגיאות. כמו כן, גורמים כמו הרקע וגוון הלבוש של הנבדק ישפיעו על טיב הזיהוי.

בבית עיבדנו את הקובץ בכמה דרכים: תחילת ביצעו עיבוד מקדים לפריים המכול הפיקת הפריים לגוני אפור, יצירת תמונה ביןארית וחיתוך של האזור הרלוונטי (ROI). לאחר מכן מצאו את מיקומי המרקרים בפריימים השונים וחישבו את המהירות של כל מפרק בכל מקטע, את המפרק הכלל שהרגל עברה וכן את זווית הירך והברך לאורך המקטעים השונים. נצפה כי מהירות המפרק תהיה גבוהה יותר ככל שהוא ממוקם נמוך יותר ברgel הנבדק בשל אופי תנועת הרגל בכך שהחלק התיכון יעבור מפרק רב יותר בזמן זהה לחלקים העליונים של הרגל [5]. כמו כן, נצפה כי החישוב שיתקבל למרחק הכלל שהרגל עברה יהיה שווה לערך לדרך הכלול של ההליכון עד כדי שגיאה מסוימת התנבע מספר סיבות; בשל שגיאות זיהוי של המפרק וכן מכיוון שתנועת הרגל אינה ישירה ומתנהגת כפרבולת כלומר מעבור מרחק רב יותר. בסוף, בשל אופי התנועה, נצפה כי זווית הירך תנעה בין 0 מעלות (מצב עמידה ישירה) לבין זווית הקטנה מכ-60 מעלות וכי זווית הברך תהיה בשינוי של עד כ-70 מעלות מזבבה ההתחלתית שהוא 180 מעלות. כמו כן, נצפה כי זוויות הברך והירך ינעו בצורה מ חוזרת עקב מחזוריות ההליכה של הנבדק.

3.4.2 מתודולוגיה

כלים וחומרים: סרטון וידאו, תוכנת המטלב.

מהלך הניסוי: ראשית ווידאנו כי המכלה ממוקמת כך שתזהה את ההליכון וניתן לזהות את האובייקטים بصورة טובה. מיקמו 4 מרקרים על רגל המשתנה במיקומים שונים לאורך הרגל, 2 מעל הברך ו-2 מתחת לברכ. צילמו וידאו של כ-70 שניות בקצב הליכה של 2 קמ"ש. השתמש בVIDAO זה לניתוח תנועת רגלי אדם במהלך ביצוע פעולה מוטורית.

על מנת למצוא את מיקומי המרקרים בפריימים השונים, ביצעו מספר שלבים. תחילת ביצעו עיבוד מקדים לפריים המכול הפיקת הפריים לגוני אפור, יצירת תמונה ביןארית עם סף של 0.4 הנבחר بصورة אמפירית וחיתוך האזור הרלוונטי (ROI). בסוף, השתמשנו במסנן חציו בגודל 3x3 לניקוי רעשים מהתמונה. לאחר מכן, השתמשנו בפונקציה `imfindcircles` תוך הגדרת פרמטרים שונים, לזיהוי המרקרים הממוקמים לאורך רגלי הנבדק, זאת בשל צורתם העגולה.

3.4.3 תוצאות

תשובה לשאלת 4.1.1:

נשרטט דיאגרמת בלוקים עבור האלגוריתם שבנוינו למציאת מיקומי המרקרים על רגלי הנבדק:

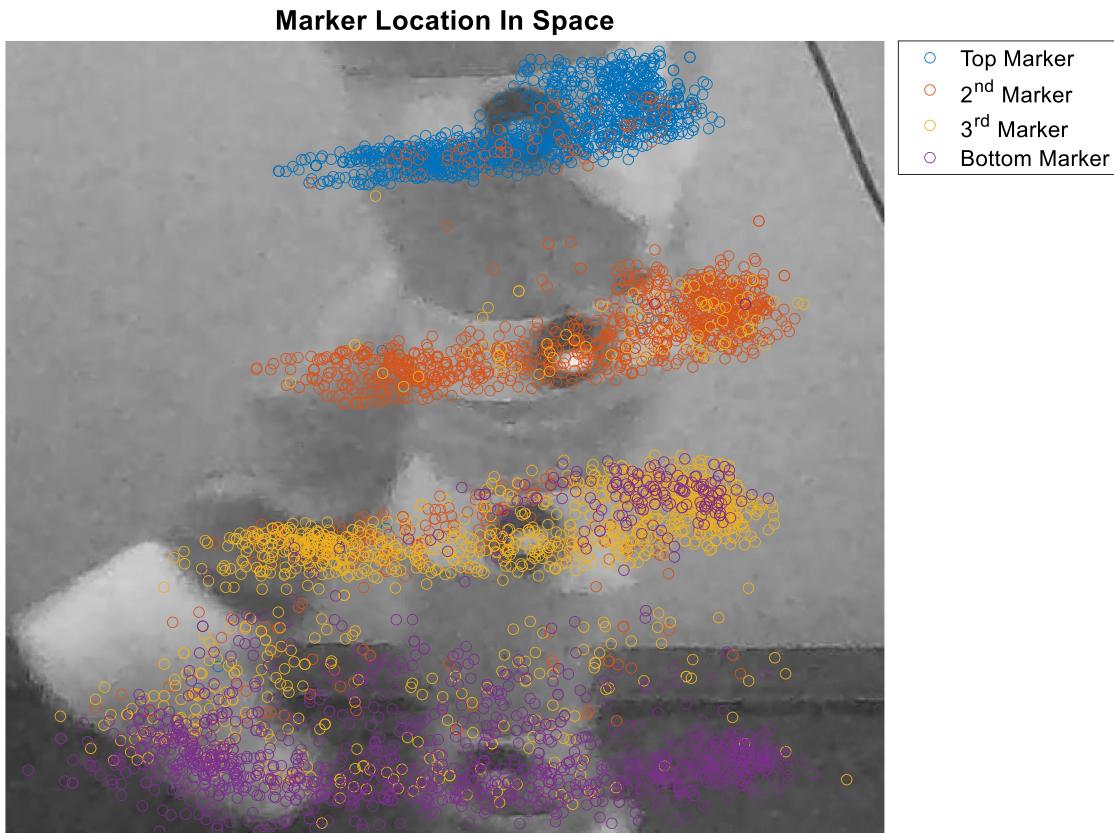


איור 24: דיאגרמת בלוקים של האלגוריתם שבנוינו

באיור 24 ניתן לראות את דיאגרמת הבלוקים למציאת מיקומי המרקרים על רגליו של הנבדק. התהליך מתחילה בהעלאת סרטון והמרת הפרייםים שלו לגוני אפור. לאחר מכן, הפרייםים מומרים לתמונהニアרית, כאשר נקבע סף של 0.4 באופן אמפירי, כדי להפריד בין האובייקטים לרקע. השלב הבא הוא שיעור הפרייםים לאזור הרלוונטי (ROI) בלבד, ככלומר אזור הרגליים של הנבדק על מנת להימנע מזיהוי אובייקטים לא רלוונטיים, צעד זה מאפשר להתקדם בחלק החשוב של התמונה ולהפחית הפרעות מיותרות. לאחר מכן, משתמשים במסנן חציו בגודל 3x3 לניקוי רעים מהתמונה, שיטה ייעילה לשמרה על חדות התמונה תוך הסרת רעים. השלב הבא שימוש בפונקציית imfindcircles, עם קבועה פרמטרים שונים (כגון טווח רדיוס המרקר אשר נקבע באמצעות שימוש בפונקציית imdistline), זיהוי אובייקט בהיר/כהה, וגישה הזיהוי ועוד) למציאת המרקרים לאורך רגלי הנבדק, שלב קריטי ליזיהוי האובייקטים הרצויים בתמונה. לאחר מציאת המרקרים, מתבצעת בחירה של ארבעת המיקומים הראשונים שזווחו ומיונים על פי ציר Y, כדי לקבוע את הסדר שלהם בצורה מדוקיקת. לבסוף, שמירת ערכי המיקומים בקואורדינטות (y,x) מאפשרת להשתמש בתנאים אלו למעקב אחר פרמטרים נוספים בהמשך, כגון מהירות הבדיקה.

תשובה לשאלת 4.1.2 :

מציג כעט גראף של מיקום המרקר במרחב :



איור 25 : מיקום המרקרים במרחב

באיור 25 ניתן לראות את מיקום המרקרים במרחב. המרקרים מחולקים ל-4 צבעים ; מרker עליון (כחול), מרker שני מלמעלה (כתום), מרker שלישי מלמטה (צהוב) ומרker תחתון (סגול). נשים לב כי המרker הכהול והכתום נעים במרחב בתחום סגור, פיזור המרקרים מרכז יחסית ונע יותר בכיוון ציר ה- x מאשר ציר ה- y . לעומת זאת, המרker הצהוב והסגול נעים במרחב גדול יותר ופיזור המרקרים יותר מפוזר, אך התבנית הכללית של הפיזור דומה לתבנית של שני המרקרים הראשונים, תנואה יותר בכיוון ציר ה- x מאשר ציר ה- y . נשים לב כי פיזור המרקרים וה坦ועה של כל מרker עולה ככל ששיעור המרker יורדת במקומו על הרגל.

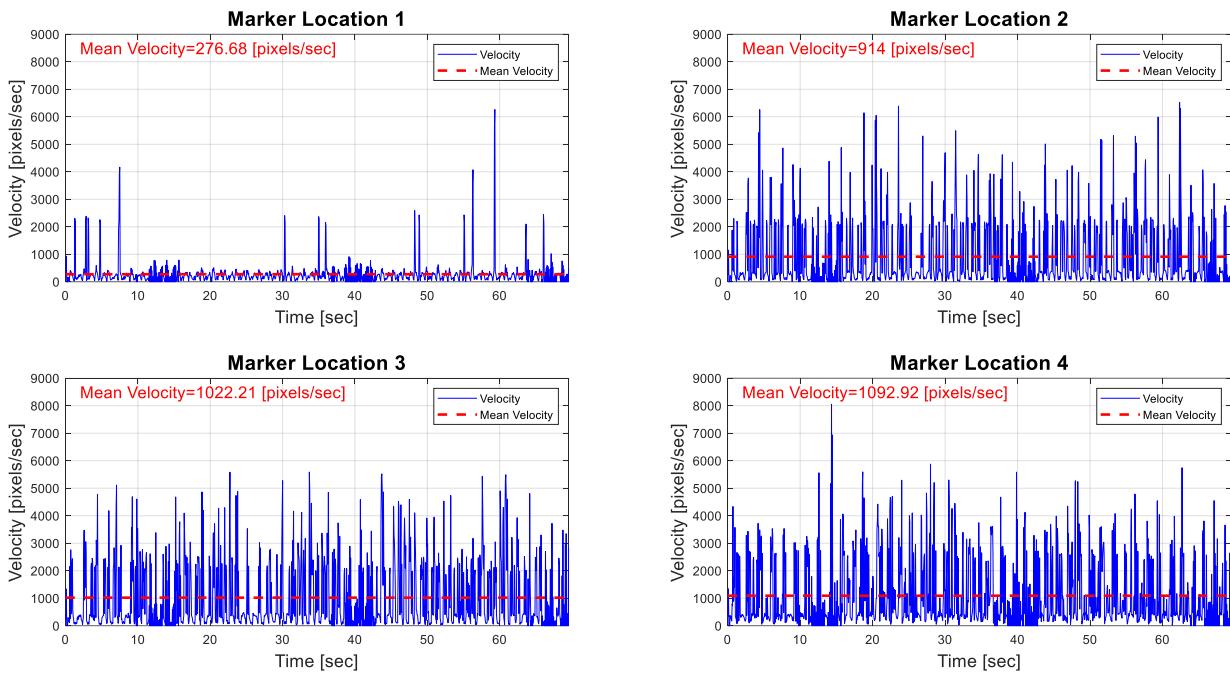
תשובה לשאלת 4.2 :

נמצא את מהירות הרגל בערך מוחלט בכל נקודות זמן. נחשב זאת באמצעות שיעורו המהירות לפי מרחק בין פרויימים סמוכים חלקי הזמן. עבור כל מרker נחשב את המרחק האוקלידי לפי נוסחה (2) בין כל 2 פרויימים סמוכים. על מנת לחשב את המהירות, נחלק את המרחק שקיבלו באורך כל פרויימ. זאת נעשה על פי הנוסחה הבאה [6] :

$$(3) \quad \bar{v} = \frac{\Delta s}{\Delta t}$$

כאשר \bar{v} הינו מהירות הממוצעת, s הינו שינוי במקומות ו- t זהו משך השינוי במקומות.

מציג גרפ של מהירות בערך מוחלט כתלות בזמן :



איור 26 : גרפ של מהירות הרגל כתלות בזמן

באיור 26 ניתן לראות את גרפי המהירות כפונקציה של הזמן, כאשר כל אחד מייצג מיקום שונה של מפרק על רגל הנבדק. מפרק 1 הינו המפרק העליון הממוקם בירך, מפרק 2 ממוקם מעל הברך, מפרק 3 ממוקם מתחת לברך ומפרק 4 הינו המפרק התיכון הממוקם בחלק התיכון של השוק. הגרפים מראים את המהירות של המפרקים במהלך הזמן, כאשר הציר האופקי מייצג את הזמן בשניות והציר האנכי מייצג את המהירות ביחידות של פיקסלים לשנייה. הקו המכוון האדום מייצג את המהירות הממוצעת לכל מפרק, והוא מאפשר השוואת נוחה בין המהירות הממוצעות של המפרקים השונים. לעומת מפרק 1, הממוקם בחלק העליון ביוטר של הרגל, ניתן לראות שהמהירות הממוצעת היא הנמוכה ביוטר הממוצע, וכך כל שמיוקם המפרק נמוך יותר (בכיוון הרגל) איזי ניתן לראות עליה במאה (276.68 [pixels/sec]) וכן הממוצעות הממוצעות הגבוהה ביוטר מתתקבל עבור מפרק 4 (1092.92 [pixels/cm]). בנוסף, הנתונים מציגים פיזור רחב של ערכי המהירות לכל מפרק, מה שעשוי להצביע על השגיאות בסיווג וכן גורמים נוספים המשפיעים על התנועה, כגון תנודות בקצב התנועה של הנבדק או חוסר יציבות במהלך הניסוי.

תשובה לשאלת 4.3 :

את המפרק הכלול שהרגל עברה בעת הליכה נוכל לחשב על ידי מדידת המרחק הכלול שהמרקף התיכון עבר. נחשב זאת על ידי סכימת כל הפרשי המרחקים שהשכנו בסעיף הקודם. בעת נרצה להמיר את המפרק מפיקסלים לסנטימטרים, זאת נעשו על ידי מציאת יחס $\frac{cm}{pixel}$. ידוע כי גודל הcadour של המפרק הינו 1 סנטימטר ולכן על ידי מדידת מספר הפיקסלים המכילים את גודלcadour (בעזרת הפונקציה imdistline בצורה ידנית) נוכל למצוא את היחס הניל. על ידי כפילה של היחס במרקף הפיקסלים שמצאנו, נקבל כי המפרק הכלול שהרגל עברה הינו $3163.34 [cm]$.

נשווה את המפרק שקיבלנו למפרק הליכה האמיתי. נחשב מפרק זה לאחר העברת אגפים בנוסחה (3) והමרת יחידות :

$$2 \left[\frac{km}{h} \right] * \frac{10^5}{1} \left[\frac{cm}{km} \right] * 69.52 [sec] * \frac{1}{3600} \left[\frac{h}{sec} \right] = 3862.29 [cm]$$

תשובה לשאלה 4.4 :

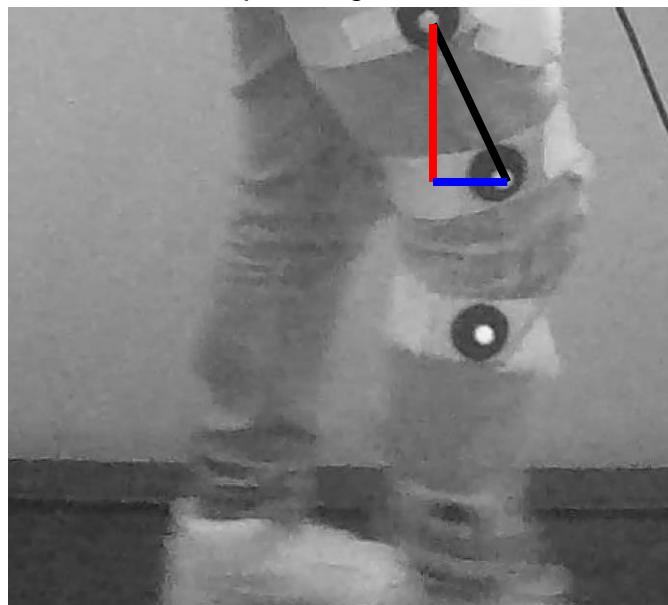
כעת נחשב את זווית הירך בכל שלב בתמונה. את הזווית נחשב בעזרת יחסים טריגונומטריים של משולש ישר זווית [7].

$$(4) \quad \sin(A) = \frac{h}{b}$$

כאשר A הינה הזווית הרצiosa לחישוב, h הינו צלע המשולש ממול לזווית A ו-b הינו היתר של המשולש ישר הזווית.

נציג את המשולש הבא על גבי הירך :

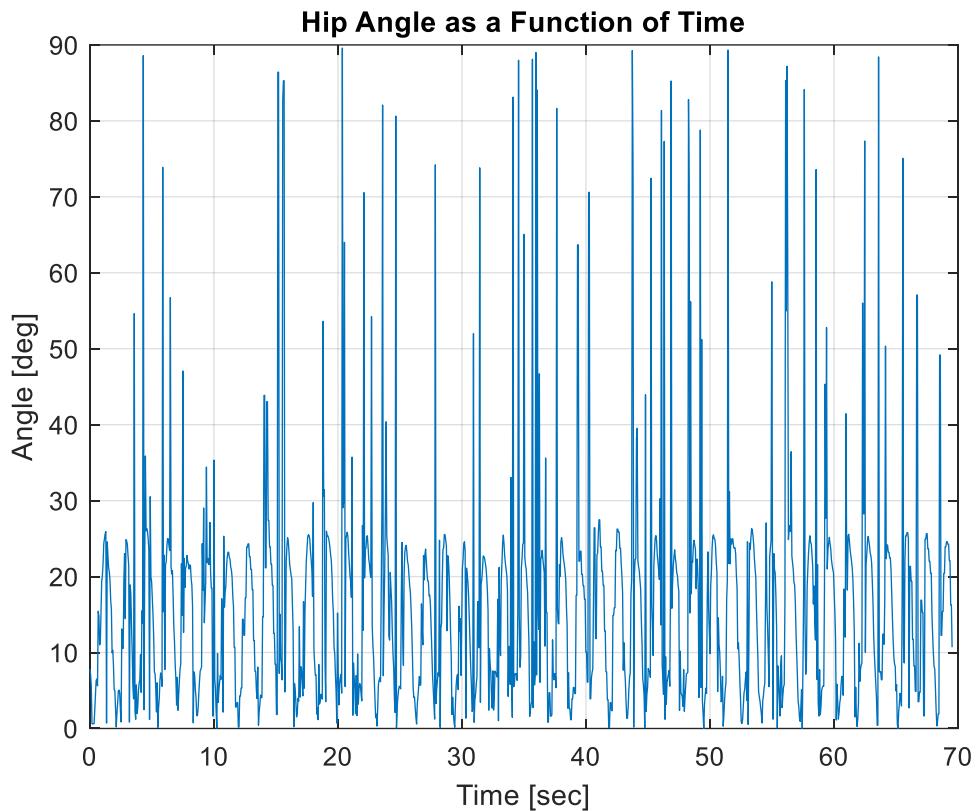
Example of Angle Detection



איור 27 : דוגמא למציאת זווית הירך בעזרת משולש ישר זווית

באיור 27 ניתן לראות את המשולש הישר זווית שהשתמשנו בו על מנת למצוא את זווית הירך. זווית הירך היא הזווית העליונה, בין הקו האדום (הנורמל לרכפה) והקו השחור (המרחק בין 2 המפרקים העליונים). נחשב לפי נוסחה (4) כאשר h הינו הקו הכהול ו-b הינו הקו השחור. בדומה זו נחשב את הזווית A ביחס לנורמל לרכפה (הקו האדום).

נציג גרף המતאר את השתנות זווית הירך כתלות בזמן :



איור 28 : נרף השתנות זווית הירך כתלות בזמן

באיור 28 ניתן לראות את גרף זווית הירך כתלות בזמן. נתיחס לזוויות 0 כאשר הנבדק עומד בזרה ישירה. נשים לב כי זווית הירך נעה בין 0 ל-90 מעלות, כאשר רוב הזוויות נעות בין 0 לכ-30 מעלות. בנוסף, נבחן כי השתנות הזווית הינה מחזורית, עד כדי התעלמות מהפיקים החדים, כאשר הזווית מתחליה מ-0 מעלות, מגיעה לפיק מסויים וחוזרת חזרה לכ-0 מעלות.

תשובה לשאלת 4.5 :

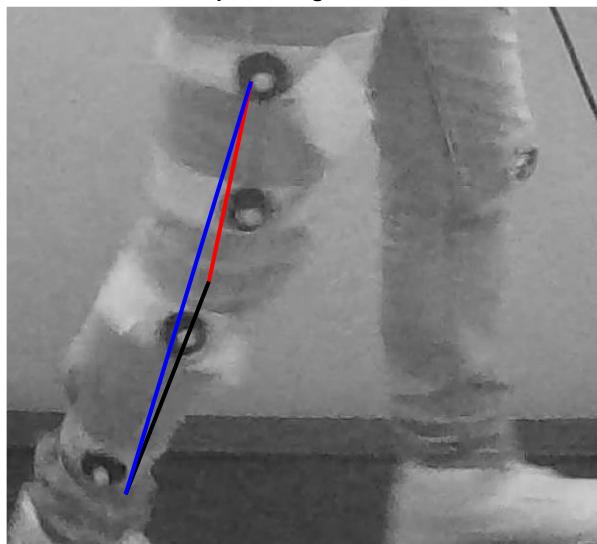
כעת נחשב את זווית הברך בכל שלב בתמונה. את הזווית נחשב בעזרת חוק הקוסינוסים המוחושם בדרך הבאה [8] :

$$(5) \quad c^2 = a^2 + b^2 - 2ab \cos(C)$$

כאשר C הינה הזווית הרצiosa לחישוב, c הינו צלע המשולש ממול הזווית הרצiosa C ו- a ו- b הינם צלעות המשולש האחרות.

מציג את המשולש הבא על גבי הברך :

Example of Angle Detection



איור 29 : המשולש על גבי הברך על מנת לזהות את זווית הברך

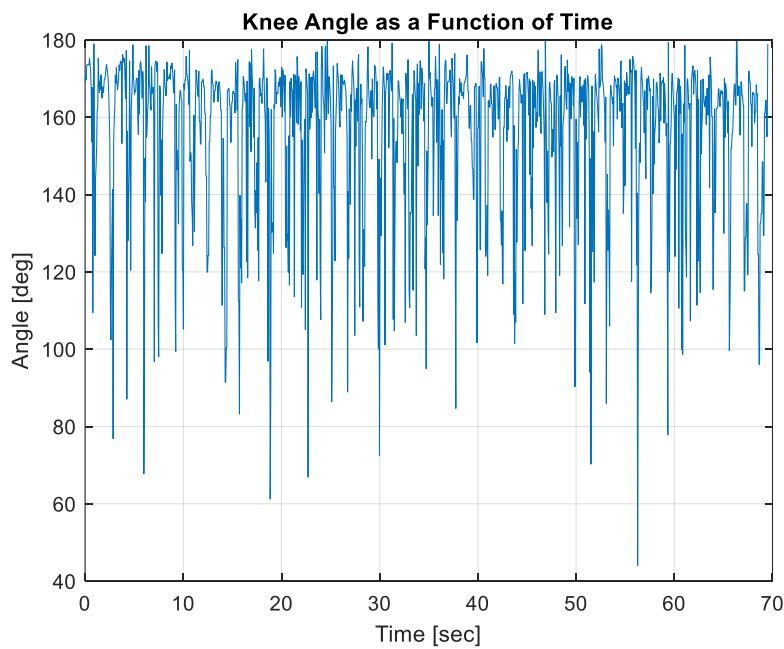
באיור 29 ניתן לראות את המשולש בו שהשתמשנו בו על מנת למצוא את זווית הברך. זווית הברך היא הזווית האמצעית בגובהה, בין הקו האדום ולקו השחור. את נקודת החיתוך בין שני קווים אלו חישבנו על ידי חישוב אמצע קו בצורה הבאה [6] :

$$(6) \quad M = \left(\frac{X_A + X_B}{2}, \frac{Y_A + Y_B}{2} \right)$$

כאשר M הינה נקודת אמצע הקו, A ו- B הינו נקודות הקצה ו- X -ו- Y הין קורדינטות כל נקודה.

נחשב את זווית הברך לפי נוסחה (5) כאשר C הינו הקו הכחול ו- a ו- b הינם הקווים השחור והאדום. בצורה זו נחשב את הזווית C שהינה הזווית שמולו הקו הכחול.

מציג גרף המתאר את השתנות זווית הברך כתלות בזמן :



איור 30 : גраф השתנות זווית הברך כתלות בזמן

באיור 30 ניתן לראות את גך זווית הברך כתלות בזמן. נקבעו לזווית 180 לזוויות כאשר הנבדק עומד בצד ימין. נשים לב כי זווית הברך נעה בין 40 ל-180 מעלות, כאשר רוב הזווית נעות בין 120 ל-180 מעלות. בנוסף, נבחין כי השתנות הזווית הינה מחזורית, כאשר הזווית מתחילה מ-180 מעלות, מגיעה לפיק מסויים (מינימום מקומי) וחוזרת חזרה ל-180 מעלות.

3.4.4 מסקנות

בניסוי זה ביצעו ניתוח של תנועת רגלי הימנית של הנבדק, באמצעות 4 מרקרים שמוקמו לאורך רגלו הימנית במיקומים שונים (2 מעל לברך ו-2 מתחת לברך). בעזרת הפונקציה שבנינו לוייה המרקרים, מצאנו את מיקומם וביצעו חישוב של מהירות, מרחק, זווית הירך וזווית הברך של הנבדק.

מציאת מיקומי המרקרים ה证实עה תוך שימוש בפרייםים, חיתוך כל פריים כך שיتمקד ברגל (ROI) ועיבוד מקדים של התמונה אשר כלל המרת התמונה לגוני אפור, מציאת סף ביןاري להבלטת המרקרים וסינוו רושים בעזרת מסנן חציון. השתמשנו בפונקציה `imfindcircles` למציאת המרקרים (צורות עגולה וצבעם לבן) ושמרנו את המיקומים. ראיינו כי בהתאם לציפויינו, המהירות של כל מפרק גדלה ככל שמיומו נמוך יותר, דבר זה הגינוי מכיוון שבתנוחת הרגל החלק התחתון עבר מרחק גדול יותר בהשוואה לחלק העליון בפרק זמן זהה [5]. בנוסף, ראיינו כי המרחק הכללי שהרגל עברה, כפי שנמדד בניסוי, הוא 3163.34 סנטימטרים, בעוד שהמרחק האמיתי שנמדד היה 3862.29 סנטימטרים. הפרער בין שני המדדים תואם לציפויינו כי יתקבלו שגיאות במהלך המדידה, וסביר כי ערך המרחק שהתקבל גדול יותר בשל שגיאות בזיהוי. בנוסף, הנתונים המתפללים מהגרפים של זווית הירך והברך מצביעים על פיזור רחב יחסית ופיקים שחורגים מהערכים הצפויים, דבר המחזק את קיומן של שגיאות בזיהוי המרקרים במהלך הניסוי.

באיור 25 נראה כי הפונקציה שבנינו לזיהוי המיקומים הייתה במרבית המקרים. הסקנו כי לקבלת תוצאות טובות ואמינות יותר יש לדאוג שהרקע וגוון הלבוש של הנבדק יהיו אחידים ובגון מתאים כדי שייהיו הפרעות מינימליות ולא יפגמו בטיב הזיהוי. בנוסף, ראיינו כי התקבלו פרייםים רבים בהם איקות התמונה הייתה ירודה (טשטוש) והתרחשו שגיאות בזיהוי הסטנסים. מכך נסיק כי ישנה חשיבות לאיכות המצלמה, יציבותה, מיקומה ותאורה מתאימה. כמו כן, ישנה חשיבות רבה לבחירת מסיכה נכונה. שינויים מזעריים יחסית בبنית המסכה (בחירה סף) השפיעו על הזיהוי באופן ניכר. כמו כן, ישנים פרמטרים רבים, אשר שינו, הוספנו והתאמנו אותם לצורה המיטבית שהצלחנו, אך סביר כי ניתן יהיה לשנות פרמטרים אלו ולהוסיף אחרים על מנת להגיע למצב אידיאלי יותר וקבלת מסכה איקוטית יותר. לכן, ישנו צורך בשיפור טיב המסכה והכרת טכניות נוספות.

4 מסקנות כלליות

במעבדה זו הכרנו את מבנה התמונה ותכונותיה במקומות ובתדר. התנסינו בסינון תמונה בדרכים שונות, ביצוע פעולות מורפולוגיות, סגמנטציה ויזיהו גבולות, מצוי מאפייני אובייקטים בתמונה וכן עיבוד וידאו.

בניסוי 1 עבדנו עם תמונה של אוזן אליה התייחסנו כתמונה טסיות. מטרתנו הייתה ליצור תמונה המורכבת אך ורק מטסיות הנוטות ימינה ('). ביצענו זאת באמצעות סגמנטציה ומיצוי מאפיינים של טסיות הדם. רأינו כי הסרת הרקע שיפרה את יכולות הסגמנטציה ואפשרה יצירה מסכה מדויקת יותר ליזיהו האובייקטים הרצויים. דבר זה הציב על כך שעיבוד מקדים של תמונה עשוי להיות בעל חשיבות רבה בהצלחת הליך הסגמנטציה. בנוסף, השתמשנו בפילטרים ושיטות שונות ליזיהו גבולות. רأינו כי שיטות כמו LoG ו-dilation יכולות להיות יעילות ליזיהו גבולות האובייקטים. עם זאת, ניתן כי במקרים מסוימים פועלות אלו עלולות לשנות את גבולות האובייקטים ולגרום ליזיהו שגוי של אובייקטים סמכים אחד. כמו כן, רأינו כי ניתן להשתמש במאפיינים כמו אוריננטציה כמו אוריינטציה והפרדה של אובייקטים לפי כיווניות מסוימת. שיטה זו נמצאה כיעילה בסינון אובייקטים בתמונה לפי תכונה גאומטרית מסוימת.

בניסוי 2 ביצענו סינון של רעש מלך פלפל מהתמונה בעזרת מסנני חציון בגדים שונים. רأינו כי סוג מסנן החציון אפקטיבי לסינון רעש מסווג זה וכי גודל הגרעין משפיע על התוצאה המתקבלת. קיבלנו כי גרעין בגודל 3x3 אופטימלי במקרה זה לשימירה על חדות התמונה תוך סינון הרעש (אייזון בטרייד אוף) והניב ערך מינימלי עבור מדד MSE. השתמשנו בניתוח סטטיסטי לבחינת עקבות התוצאה. קיבלנו כי קיים הבדל מובהק בין מסנני החציון הריבועיים השונים. רأינו כי חזרות ניסויים וניתוחים סטטיסטיים יכולים להצביע על עקבות התוצאות ולהבדיל בין גדים שונים של מסננים. דבר המחזק את מוצקות התוצאות.

בניסוי 3 סיוגנו את כיווני ההסתכלות של הנבדק ל-4 אזורי הסתכלות. ביצענו זאת תוך שימוש בתמונות כיוול, מציאת מיקומי העיניים בתמונות אלו ואפיון מיקומי העיניים. בניתוח הסרטון ביצענו התאמות של התמונה כך שתתאים למטריצת הכיוול שבינו, זאת בשל שינוי במנת הפנים של הנבדק בסרטון בהתאם לתמונות הכיוול. רأינו כי להטאה נכונה לתמונות הכיוול ישנה חשיבות רבה באחיזה ההצלחה של ליזיהו אזור ההסתכלות. בנוסף, קיבלנו כי הסיוג בעזרת עין ימין סיפק תוצאות מדויקות כగון תאורה אשר הקשו על ליזיהו של מיקום עין שמאל במקרים רבים, אשר סברנו כי נובעת בשל שינויים כגון תאורה אשר השפיעו על ליזיהו מיקום עין שמאל בצורה טובה. לכן, במקרה זה נעדיף להסתמך על עין ימין בסיווג אזור ההסתכלות, אשר גם בבדיקה תמונות הכיוול הניבה תוצאה טובה יותר. כמו כן, רأינו כי חיפפת אזוריים עלולה לגרום לשגיאות סיוג. עבור עין ימין קיבלנו הפרדה טובה למדי בין אזורי ההסתכלות השונים.

בניסוי 4 ניתחנו סרטון של תנועת רגלו הימנית של הנבדק בעזרת מרקרים שמוקמו לאורך הרגל בגבהים שונים. אפיון המידע כלל את מיקום המרקרים, מהירות, זווית הברך וזווית הירך. נמצא כי המהירות של כל מפרק על הרגל גדולה ככל שמיומו נמוך יותר, דבר התואם לציפויו מתנועת הרגל. כמו כן, רأינו כי המרחק הכלול שהרגל עברה, כפי שנמדד בניסוי, הוא 3163.34 סנטימטרים, בעוד שהמרחק האמצעי שנמדד היה 3862.29 סנטימטרים. הופיע בין שני המדדים תאם לציפויו לנוילט שגיאות בתהיליך המדידה, והיה סביר כי ערך המרחק שהתקבל גדול יותר בשל שגיאות בזיהוי. הנתונים שהתקבלו בגרפים של זווית הירך והברך הציגו פיזור רחב יחסית ופיקים שחורגים מהערכים הצפויים, דבר אשר חזק את קיומו של שגיאות בזיהוי המרקרים במהלך הניסוי. כמו כן, רأינו כי יכולות הזיהוי של המרקרים תלויות באיכות הרקע ובגונו של הלבוש של הנבדק. רקע ולבוש אחידים ובגונו מתאים יכולים לשפר את טיב הזיהוי. ניסוי זה הדגים את החשיבות של יכולות המצלמה, יציבותה, מיקומה ותאורה מתאימה בזיהוי מarkers ובניתוח תנועה. פרמטרים אלו משפיעים באופן ניכר על דיקוז היזיהו ועל איכות התוצאות הסופיות. בנוסף, רأינו כי השימוש במסנן חציון לסינון רעים ובחרית סף ביןארוי מתאים להבלטה המרקרים הוא קריטי ליזיהו נוכן של המרקרים, שכן שגיאות קטנים בבחירה המסכה יכולים להשפיע משמעותית על דיקוז היזיהו.

5 ביבליוגרפיה

- [1] J. Frost, “Mean Squared Error (MSE),” Statistics By Jim. Accessed: Mar. 26, 2024. [Online]. Available: <https://statisticsbyjim.com/regression/mean-squared-error-mse/>
- [2] “One-Way ANOVA.” Accessed: Mar. 26, 2024. [Online]. Available: https://www.jmp.com/en_be/statistics-knowledge-portal/one-way-anova.html
- [3] “Image Coordinate Systems - MATLAB & Simulink.” Accessed: Mar. 26, 2024. [Online]. Available: <https://www.mathworks.com/help/images/image-coordinate-systems.html>
- [4] D. Cohen, *Precalculus: A Problems-Oriented Approach*. Cengage Learning, 2004.
- [5] J. M. Winters and S. L.-Y. Woo, Eds., *Multiple Muscle Systems: Biomechanics and Movement Organization*. New York, NY: Springer, 1990. doi: 10.1007/978-1-4613-9030-5.
- [6] “The Feynman Lectures on Physics Vol. I Ch. 8: Motion.” Accessed: Mar. 26, 2024. [Online]. Available: https://www.feynmanlectures.caltech.edu/I_08.html
- [7] “2.1: The Law of Sines,” Mathematics LibreTexts. Accessed: Mar. 26, 2024. [Online]. Available: [https://math.libretexts.org/Bookshelves/Precalculus/Elementary_Trigonometry_\(Corral\)/02%3A_General_Triangles/2.01%3A_The_Law_of_Sines](https://math.libretexts.org/Bookshelves/Precalculus/Elementary_Trigonometry_(Corral)/02%3A_General_Triangles/2.01%3A_The_Law_of_Sines)
- [8] “2.2: The Law of Cosines,” Mathematics LibreTexts. Accessed: Mar. 26, 2024. [Online]. Available: [https://math.libretexts.org/Bookshelves/Precalculus/Elementary_Trigonometry_\(Corral\)/02%3A_General_Triangles/2.02%3A_The_Law_of_Cosines](https://math.libretexts.org/Bookshelves/Precalculus/Elementary_Trigonometry_(Corral)/02%3A_General_Triangles/2.02%3A_The_Law_of_Cosines)
- [9] “Midpoint of a Line Segment.” Accessed: Mar. 26, 2024. [Online]. Available: <https://www.mathsisfun.com/algebra/line-midpoint.html>

```
% Adjusting the default figure settings
set(0,'defaultAxesXGrid','on')
set(0,'defaultAxesYGrid','on')
print=0;
%% EXP 1
% 1.1
% Loading blood platelets picture
rice=imread('rice.png');
rice=mat2gray(rice);
% Showing the picture
figure
imshow(rice)
title('Blood Platelets Photo')

% 1.2
% Plotting histogram
figure
imhist(rice)
title('Histogram of Gray Levels - Original Photo')
xlabel({'',' ',' ','Gray Levels'})
ylabel('Count')
lim=ylim;
ylim(lim+[0 500])

% 1.3
% Creating mask manually
TH=0.5;
riceMask=rice;
riceMask(riceMask>=TH)=1;
riceMask(riceMask<TH)=0;
% Showing the mask
figure
imshow(riceMask)
title('Rice Mask - Manual Threshold')

% 1.4
% Cheacking manualy what are the dimentions of the blood platlets
h=imdilate;
delete(h)
% Creating the disk structure with the smaller blood platlets radius
disk=strel('disk',10);
% Using erosion to remove the blood platelets
notRice=imerode(rice,disk);
% Showing the background
figure
imshow(notRice)
title('Background')
% Removing background
onlyRice=rice-notRice;
figure
imshow(onlyRice)
title('Only the Blood Platlets')

% 1.5
% Plotting new histogram
figure
```

```

imhist(onlyRice)
title('Histogram of Gray Levels - After Filtering')
xlabel({'',' ',' ','Gray Levels'})
ylabel('Count')
lim=ylim;
ylim(lim+[0 1500])
% New threshold
TH=0.35;
riceMaskOR=onlyRice;
riceMaskOR(riceMaskOR>=TH)=1;
riceMaskOR(riceMaskOR<TH)=0;
% Plotting new mask
figure
imshow(riceMaskOR)
title('Blood Platelets Mask - After Background Removal')

% Edge detection dilation-mask
edgeD_M=imdilate(riceMaskOR,strel('diamond',1))-riceMaskOR;
% Plotting the results
figure
nexttile
imshow(edgeD_M)
title('Blood Platelets After Dilation-Mask')
% Edge detection laplacian of gaussian
nexttile
edge(onlyRice,'log')
title('Blood Platelets After LoG')

% 1.6
% Calculating number of blood plateles
[ricEdges,nObjs]=bwlabel(imclearborder(edgeD_M));
figure
imshow(ricEdges)
title(['Clean Edge Detection - Number of Blood Plateles=' num2str(nObjs)])

% 1.7
% Finding the blood platlets with right oriantation
orientInfo=regionprops(ricEdges,'Orientation','FilledImage','BoundingBox');
RightOrientInfo=orientInfo([orientInfo.Orientation]>0);
% Creating The mask. Starting with blank photo
RightMask=zeros(size(rice));
for i=1:length(RightOrientInfo)
    % Indexes for up left corner
    LeftCornerX=ceil(RightOrientInfo(i).BoundingBox(1));
    LeftCornerY=ceil(RightOrientInfo(i).BoundingBox(2));
    % All the indices for the right orianted blood platletes
    indX=LeftCornerX:LeftCornerX+RightOrientInfo(i).BoundingBox(3)-1;
    indY=LeftCornerY:LeftCornerY+RightOrientInfo(i).BoundingBox(4)-1;
    % Adding back the parts containing the full right blood platlets
    RightMask(indY,indX)=RightMask(indY,indX)+RightOrientInfo(i).FilledImage;
end
% Showing the mask
figure
imshow(RightMask)
title('Only Right Orianted Blood Plateles - Mask')

% 1.8
% Applying the mask to the original photo
OnlyRightRice=rice.*RightMask;

```

```

% Showing results
figure
imshow(OnlyRightRice)
title('Only Right Oriented Blood Plateles - After Masking')

%% Exp 2
% Loading photos
RegVeg=imread('images\vegetables.tif');
RegVeg=mat2gray(RegVeg);
NoiseVeg=imread('images\Nvegetables.tif');
NoiseVeg=mat2gray(NoiseVeg);
% Showing the photos
figure
nexttile
imshow(RegVeg)
title('Vegetables')
nexttile
imshow(NoiseVeg)
title('Noisy Vegetables')

% 2.1
% Filter dimentions
var1=[1 1 2 3 3 4 4 5 7];
var2=[2 3 2 3 4 4 5 5 7];
% Allocating memory
MSE=zeros(length(var2),1);
figure(WindowState="maximized")
for i=1:length(var2)
    % Median filtering
    FilteredVeg=CleanSP(NoiseVeg, 'Median', var1(i), var2(i));
    % Calculating MSE
    MSE(i)=mean((FilteredVeg-RegVeg).^2, "all");
    % Showing results
    nexttile
    imshow(FilteredVeg)
    title({['Median Filter ' num2str(var1(i)) 'x' num2str(var2(i))], ['MSE='
    num2str(MSE(i))]}, FontSize=16)
end
% Plotting MSE graph
figure
plot(MSE)
title('MSE for Various Median Filters Sizes')
xlabel('Median Size')
ylabel('MSE')
xticklabels({'1x2' '1x3' '2x2' '3x3' '3x4' '4x4' '4x5' '5x5' '7x7'})

% 2.2
% Allocating memory
MSE=zeros(100,length(var2));
% Loop for 100 different noise and filtering simulation
for i=1:100
    % Adding salt&peper noise - 20%
    SPNoisyVeg=imnoise(RegVeg, 'salt & pepper', 0.2);
    for j=1:length(var2)
        % Filtering with all the median filters sizes
        FilteredVeg=CleanSP(SPNoisyVeg, 'Median', var1(j), var2(j));
        % Calculating MSE
        MSE(i,j)=mean((FilteredVeg-RegVeg).^2, "all");
    end
end

```

```

end
% Calculating maen and std
meanMSE=mean(MSE);
stdMSE=std(MSE);

% Plotting boxplot
figure(WindowState="maximized")
boxplot(MSE,'Labels',{'1x2' '1x3' '2x2' '3x3' '3x4' '4x4' '4x5' '5x5' '7x7'})
title('MSE boxplot',FontSize=16)
xlabel('Median Filter Size',FontSize=14)
ylabel('MSE',FontSize=14)

% Statistical test for square filteres
squareMSE=MSE(:,var1==var2);
% One-way ANOVA statistical test
[p,tbl]=anova1(squareMSE,[],'off');
disp(tbl)
disp(['p-Value=' num2str(p)])
```

%% Exp 3

% 3.1

% Load Data

```

ImageFolder='image processing lab\eyes\' ;
% Get a list of all files in the folder with the desired file name pattern
ImagData=dir(fullfile(ImageFolder,'*.jpg'));
```

lenIm=length(ImagData);

% Preallocate a cell array to hold the images

```

EyeImag=cell(lenIm, 1);
for i=1:lenIm
    % Read the current image
    imgPath=fullfile(ImageFolder,ImagData(i).name);
    img=imread(imgPath);
    img=im2gray(img);
    img=mat2gray(img);

    EyeImag{i}=img;
end
```

% 3.3

% Preallocating

```

N=lenIm/4;
CalibFrames={cell(1,N);cell(1,N);cell(1,N);cell(1,N)};
% 3.3.1
% Creating CalibFrame cell array
for i=1:lenIm
    Frame=EyeImag{i};
    Frame=imcrop(Frame,[1103.5 343.5 702 129]);

    CalibFrames{mod(i-1,4)+1}{ceil(i/4)}=Frame;
end
```

% 3.1

% Plotting an example

```

Eye_Pos=EyePosition_FUNC(Frame);
% Plotting
figure
imshow(Frame)
hold on
```

```

viscircles(Eye_Pos,28,'EdgeColor','b');
plot(Eye_Pos(1,1),Eye_Pos(1,2),'r*')
plot(Eye_Pos(2,1),Eye_Pos(2,2),'r*')
title('Eye Detection Example')

% 3.3
% Calculating the calibration matrix
CalibMat=EyeCalibration_FUNC(CalibFrames);

% 3.3.2
% 3-D Histogram of Left Eye
figure
hist3(CalibMat(:,[1 2],1),'FaceColor','magenta')
hold on
hist3(CalibMat(:,[3 4],1),'FaceColor','blue')
hist3(CalibMat(:,[5 6],1),'FaceColor','yellow')
hist3(CalibMat(:,[7 8],1),'FaceColor','white')

title('Calibration 3-D Histogram - Left Eye Position')
xlabel('X index')
ylabel('Y index')
zlabel('Count')
xlim padded
ylim padded
zlim tight
legend('1','2','3','4')

% 3-D Histogram of Right Eye
figure
hist3(CalibMat(:,[1 2],2),'FaceColor','magenta')
hold on
hist3(CalibMat(:,[3 4],2),'FaceColor','blue')
hist3(CalibMat(:,[5 6],2),'FaceColor','yellow')
hist3(CalibMat(:,[7 8],2),'FaceColor','white')

title('Calibration 3-D Histogram - Right Eye Position')
xlabel('X index')
ylabel('Y index')
zlabel('Count')
xlim padded
ylim padded
zlim tight
legend('1','2','3','4')

% 3.4.2
% Testing the calibration matrix
for i=1:4
    Eye_Look=EyeLook_FUNC(CalibFrames{i}{end},CalibMat);
    disp(['Left Eye Look Number: ' num2str(Eye_Look(1))])
    disp(['Right Eye Look Number: ' num2str(Eye_Look(2))])
    disp(' ')
end

% 3.5
% Loading eye movement video
vid=VideoReader([ImageFolder 'WIN_20240123_05_11_14_Pro.mp4']);
% Segmentation
segmentation=ceil(([0 2.09,4.26,6.23,8.27,10.25,12.30,13.98,15.95,17.91 ...
,20.06,22.01,24.10,26.30,28.04,29.31]+1.61)*vid.FrameRate);

```

```

% Calculate look duration
lookDuration=diff(segmentation);
% The number on screen look order
numOrder=[1,2,3,2,3,4,3,2,1,2,1,2,3,1,2];
numOrderFrame=repelem(numOrder,lookDuration);

% 3.6
% For removing the beginning part of now eye movement
displace=segmentation(1);
% Preallocating memory
Eye_Look=zeros(vid.NumFrames-displace,2);
for i=displace+1:vid.NumFrames
    % Extracting frame by frame
    Frame=read(vid,i);
    Frame=im2gray(Frame);
    Frame=mat2gray(Frame);

    % Finding the relevant ROI
    cropPosition=[1103.5 343.5 702 129]-[320 75 702/4 129/4];
    currFrame=imcrop(Frame,cropPosition);
    % Resizing the current frame
    targetSize=size(CalibFrames{1}{1});
    currFrame=imresize(currFrame,targetSize);

    % Applying our functions
    Eye_Look(i-displace,:)=EyeLook_FUNC(currFrame,CalibMat);
end

% Plotting confusion matrixes
figure
nexttile
confusionchart(numOrderFrame,Eye_Look(:,1))
errLeft=nnz(logical(numOrderFrame.'-Eye_Look(:,1)))/length(numOrderFrame)*100;
title(['Confusion Matrix for Left Eye - Error Percentage=' num2str(errLeft)])

nexttile
confusionchart(numOrderFrame,Eye_Look(:,2))
errRight=nnz(logical(numOrderFrame.'-Eye_Look(:,2)))/length(numOrderFrame)*100;
title(['Confusion Matrix for Right Eye - Error Percentage=' num2str(errRight)])

% Plotting comparison
figure(WindowState="maximized")
plot(numOrderFrame,LineWidth=2)
hold on
scatter(1:length(numOrderFrame),Eye_Look(:,2),Marker="x")
title('Algorithm Recognition Comparison',FontSize=16)
xlabel('Frame',FontSize=14)
ylabel('Look State',FontSize=14)
legend('True Labels','Predicted Labels Right',FontSize=13)
xlim tight
ylim padded
yticks([1 2 3 4])

%% EXP 4
% Loading the walking video
vid=VideoReader('image processing lab\WIN_20240123_14_45_03_Pro.mp4');
% 4.1
[AllLegPositionX,AllLegPositionY]=deal(zeros(4,vid.NumFrames));
for i=1:vid.NumFrames

```

```

clc
% Read frame
Frame=read(vid,i);
Frame=im2gray(Frame);
Frame=mat2gray(Frame);
% Creating a mask and cropping the ROI
currFrame=imbinarize(Frame,0.4);
currFrame=imcrop(currFrame,[589.5 380 487 440]);
% Applying median filter
currFrame=CleanSP(currFrame,'Median',3,3);
% Applying imfindcircles function
min_rad=4;
max_rad=15;
[centers,radii,metric]=imfindcircles(currFrame,[min_rad max_rad] ...
    , 'ObjectPolarity','bright','EdgeThreshold',0.7,'Sensitivity',0.88 ...
    , 'Method','TwoStage');
% Find indices
Leg_Pos=ceil(centers);

% Sorting the centers according to Y index
centers4=centers(1:4,:);
[~,Ind]=sort(centers4(1:4,2));
% Allocating centers
AllLegPositionX(:,i)=centers(Ind,1);
AllLegPositionY(:,i)=centers(Ind,2);
end
% 4.1.2
% Plotting walk representation
FrameEx=imcrop(Frame,[589.5 380 487 440]);
figure(WindowState="maximized")
nexttile
imshow(FrameEx)
% 4.3 Measuring marker size manually
h=imdistline;
delete(h)
% End
hold on
scatter(AllLegPositionX(1,:),AllLegPositionY(1,:))
scatter(AllLegPositionX(2,:),AllLegPositionY(2,:))
scatter(AllLegPositionX(3,:),AllLegPositionY(3,:))
scatter(AllLegPositionX(4,:),AllLegPositionY(4,:))

title('Marker Location In Space',FontSize=16)
legend('Top Marker','2^{nd} Marker','3^{rd} Marker','Bottom Marker' ...
    ,Location='bestoutside',FontSize=13)

% 4.2
% Find the absolute velocity
FrameTime=vid.Duration/vid.NumFrames; % Time for each frame
% Speed Calculation
N=size(AllLegPositionX,2)-1;
% Initialize matrices to store distances and velocities
[distances,velocities]=deal(zeros(4,N));

% Calculate distances and velocities
for i=1:4 % For each location
    for j=1:N % For each frame, excluding the last one
        % Calculate distance between consecutive frames
        distances(i,j)=sqrt((AllLegPositionX(i,j+1)-AllLegPositionX(i,j))^2 ...

```

```

        +(AllLegPositionY(i,j+1)-AllLegPositionY(i,j))^2);
    velocities(i,j)=distances(i,j)/FrameTime;
end
% distances matrix now contains the distance between each consecutive frame for
% each location
% velocities matrix contains the velocity in pixels/frame for each location

% Absolute velocities
AbsVelocities=abs(velocities);
% Mean Velocity Calculation
MeanVel=mean(AbsVelocities,2);

% Assuming velocities is a 4xN matrix with the velocities for each location
% Assuming MeanVel is a 4x1 vector with the mean velocities for each location
% Assuming FrameTime is the time for each frame

% Calculate the time points for the x-axis
time=linspace(0,vid.Duration,N);
% Create subplots for each location
figure(WindowState="maximized")
for i=1:4
    subplot(2,2,i)
    plot(time,velocities(i,:),'b') % Plot absolute velocity for location i
    hold on
    plot(time,MeanVel(i)*ones(size(time)), 'r--','LineWidth',2) % Plot mean
velocity line for location i
    text(2,8500,['Mean Velocity=' num2str(round(MeanVel(i),2)) ' '
[pixels/sec]',color='r',FontSize=13)

    % Add title and labels
    title(sprintf('Marker Location %d',i),FontSize=16)
    xlabel('Time [sec]',FontSize=14)
    ylabel('Velocity [pixels/sec]',FontSize=14)
    xlim tight
    ylim([0 9000])
    % Add legend
    legend('Velocity','Mean Velocity')
end

% 4.3
% Converting from pixels to cm
MarkerSize=1; % [cm]
PixelMarkerSize=24; % [Pixels]
% Ratio
ratioPixel2Cm=MarkerSize/PixelMarkerSize;
% Calculating cumulative distance according to the bottom marker in cm
walkDist=sum(distances(4,:))*ratioPixel2Cm;
realWalkDist=2*vid.Duration*1/3600*10^5;
disp(['Walking Distance = ' num2str(walkDist) ' [cm]'])
disp(['Real Walking Distance = ' num2str(realWalkDist) ' [cm]'])

% 4.4
% Calculating distance between the top markers
hipAngleCalcX=diff([AllLegPositionX(1,:);AllLegPositionX(2,:)])^.^2;
hipAngleCalcY=diff([AllLegPositionY(1,:);AllLegPositionY(2,:)])^.^2;

hypoten=sqrt(hipAngleCalcX+hipAngleCalcY);
% Calculating the parallel to the hypotenuse

```

```

para=sqrt(hipAngleCalcX);

% Using a right triangle trigonometric ratio, calculate the apex angle
hipAngle=rad2deg(asin(para./hypoten));

% Calculate the time vector and plot the hip angle
time=linspace(0,vid.Duration,N+1);
figure
plot(time,hipAngle)
title('Hip Angle as a Function of Time')
xlabel('Time [sec]')
ylabel('Angle [deg]')

% Example of hip angle
Nframe=400;
Frame=read(vid,Nframe);
Frame=im2gray(Frame);
Frame=mat2gray(Frame);
Frame=imcrop(Frame,[589.5 380 487 440]);
% Example of angle detection
figure
imshow(Frame)
hold on
plot([AllLegPositionX(1,Nframe),AllLegPositionX(2,Nframe)], ...
      [AllLegPositionY(1,Nframe),AllLegPositionY(2,Nframe)],'k',LineWidth=3.5)
plot([AllLegPositionX(1,Nframe),AllLegPositionX(1,Nframe)], ...
      [AllLegPositionY(1,Nframe),AllLegPositionY(2,Nframe)],'r',LineWidth=3.5)
plot([AllLegPositionX(1,Nframe),AllLegPositionX(2,Nframe)], ...
      [AllLegPositionY(2,Nframe),AllLegPositionY(2,Nframe)],'b',LineWidth=3.5)
title('Example of Angle Detection')

% 4.5
% Calculating all the triangle legs distance
kneeAngleCalcX=diff([AllLegPositionX(1,:); ...
                     (AllLegPositionX(2,:)+AllLegPositionX(3,:))/2; % Middle leg calculated as the
mean between the 2 middle markers
                     AllLegPositionX(4,:); ...
                     AllLegPositionX(1,:)])).^2;
kneeAngleCalcY=diff([AllLegPositionY(1,:); ...
                     (AllLegPositionY(2,:)+AllLegPositionY(3,:))/2; % Middle leg calculated as the
mean between the 2 middle markers
                     AllLegPositionY(4,:); ...
                     AllLegPositionY(1,:)])).^2;

kneeAngleCalc=sqrt(kneeAngleCalcX+kneeAngleCalcY);
% Calculating the knee angle using the law of cosines
a=kneeAngleCalc(1,:);
b=kneeAngleCalc(2,:);
c=kneeAngleCalc(3,:);

kneeAngle=rad2deg(acos((a.^2+b.^2-c.^2)./(2*a.*b)));
% Plotting knee angle
figure
plot(time,kneeAngle)
title('Knee Angle as a Function of Time')
xlabel('Time [sec]')
ylabel('Angle [deg]')

% Example of knee angle

```

```

figure
Nframe=362;
Frame=read(vid,Nframe);
Frame=im2gray(Frame);
Frame=mat2gray(Frame);
Frame=imcrop(Frame,[589.5 380 487 440]);
% Example of angle detection
imshow(Frame)
hold on
plot([AllLegPositionX(4,Nframe),mean([AllLegPositionX(2,Nframe),AllLegPositionX(3,Nframe)])], ...
[AllLegPositionY(4,Nframe),mean([AllLegPositionY(2,Nframe),AllLegPositionY(3,Nframe)])], 'k',LineWidth=2)
plot([mean([AllLegPositionX(2,Nframe),AllLegPositionX(3,Nframe)]),AllLegPositionX(1,Nframe)], ...
[mean([AllLegPositionY(2,Nframe),AllLegPositionY(3,Nframe)]),AllLegPositionY(1,Nframe)], 'r',LineWidth=2)
plot([AllLegPositionX(1,Nframe),AllLegPositionX(4,Nframe)], ...
[AllLegPositionY(1,Nframe),AllLegPositionY(4,Nframe)], 'b',LineWidth=2)
title('Example of Angle Detection')

```

פונקציית CleanSP

```

function out_I = CleanSP(in_I, Type, var1, var2)
if strcmp(Type, 'Gaussian')
    h = fspecial('gaussian', var1, var2);
    out_I = filter2(h, in_I, 'same');
elseif strcmp(Type, 'Median')
    out_I = medfilt2(in_I, [var1 var2]);
else
    error('Invalid filter type');
end
end

```