



בשיתוף עם
אוניברסיטת תל אביב



המעבדה לעיבוד
אותות ותמונות



הפקולטה
להנדסת חשמל



הטכניון – מכון טכנולוגי
לישראל

ספר פרויקט

ניתוח נוצות התנשמת

Feather Analysis for the Barn Owl

נפתלי קיזנר	מבצע
יובל בהט ואורי בריט	מנחים
אורי פלג ומוטי צ'רטר, המחלקה לזואולוגיה באוניברסיטת תל אביב	בשיתוף עם
חורף 2012	סמסטר רישום
ספטמבר 2012	תאריך הגשה



תוכן עניינים

3	תקציר
3	פרק 1: מבוא
5	1.1 מטרות הפרויקט
5	1.1.1 חילוץ פרמטרים
5	1.1.2 עמידות
6	1.1.3 דיוק
6	1.1.4 ממשק משתמש
7	פרק 2: רקע טכני
7	2.1 קורלציה
7	2.2 LUT
8	2.3 הרחבה וכרסום
9	2.3.1 הרחבה וכרסום בגווי אפור
10	2.4 אלגוריתם k-means
12	פרק 3: תיאור כללי
12	3.1 זיהוי כרטיס כיול
13	3.2 כיול
13	3.2.1 קנה מידה
13	3.2.2 צבעים
13	3.3 זיהוי נקודות
14	3.4 הוספת נקודה נבחרת
14	3.5 הפקת נתונים
15	פרק 4: מימוש
15	4.1 זיהוי כרטיס כיול
15	4.1.1 מיקום
17	4.1.2 גודל וזווית
19	4.2 כיול
19	4.2.1 קנה מידה
19	4.2.2 צבעים
21	4.3 זיהוי נקודות
23	4.4 הוספת נקודה נבחרת
25	4.5 הפקת נתונים
27	פרק 5: ממשק משתמש
31	5.1 פתרונות מיוחדים
33	פרק 6: תוצאות
37	פרק 7: מסקנות
39	פרק 8: סיכום
39	8.1 פיתוח עתידי
41	ביבליוגרפיה

תקציר

התנשמת הלבנה היא עוף דורס לילי נפוץ ביותר. מחקר שנעשה במחלקה לזאולוגיה של אוניברסיטת תל אביב בוחן את הנקודות השחורות על גחונה. בפרויקט זה, נבנתה מערכת עבור המחקר שמקבלת תמונות של התנשמות ובוחנת את הנקודות באופן אוטומטי למחצה: היא מייילת לבד את צבעי התמונה ואת קנה המידה שלה, ולאחר מכן מנתחת את הנקודות לפי אזורי שמסמן המשתמש. כיול התמונות מתבצע בעזרת כרטיס כיול שמופיע בכל תמונה לצד התנשמת, ולו צורה טיפוסית שמזוהה ע"י קורלציה. כיול הצבעים מתבצע באמצעות טבלת התאמה לינארית. זיהוי הנקודות מתבצע באמצעות שימוש בפעולה המורפולוגית bottom hat שמבודדת את הנקודות מהרקע, ולאחר מכן הפעלת אלגוריתם k-means שמסווג את הפיקסלים לנקודות ולפלומה.

פרק 1: מבוא

התנשמת הלבנה היא עוף דורס לילי נפוץ ביותר. אזור המחיה שלה סמוך להתיישבות בני אדם לעיתים קרובות, לרבות אזורים חקלאיים. התנשמת ניזונה ממכרסמים, ולכן מהווה גורם מדביר טבעי עבור בני האדם. לתנשמת נוצות בצבע חום זהוב, וגחונה לבן ומנוקד בנקודות שחורות.

מחקר שנעשה במחלקה לזאולוגיה של אוניברסיטת תל אביב מבקש למצוא התאמה בין מאפיינים ויזואליים של התנשמת לבין תנאי המחיה שלה. באופן יותר ספציפי, המחקר רוצה לדעת האם דפוס הנקודות שעל גחון התנשמת מקבל את מראהו כתוצאה מסוג המזון של התנשמת, טפמרטורת המחיה שלה, אזור המחיה שלה וכו'.

השיטה המסורתית של איסוף הנתונים היא לכידת התנשמת, הצמדתה פרקדן לשולחן, צילום איזור החזה והבטן שלה, ולבסוף ניתוח ידני של התמונות שהתקבלו. בתמונות מופיעות לעיתים עשרות ומאות נקודות אליהן יש להתייחס, וניתוח ידני של כזו כמות של מידע לוקח זמן רב.

בפרויקט זה נבנתה מערכת לעיבוד ממוחשב של התמונות, על מנת לחסוך לחוקרים זמן יקר. המערכת היא אוטומטית למחצה, כלומר אין המערכת אמורה לזהות את התנשמת בתמונה, אלא אזור העיבוד ייקלט מהמשתמש. עם זאת, ישנו שלב שהינו אוטומטי לחלוטין, והוא כיול צבעי התמונה ומציאת קנה המידה שלה.



איור 1 - תנשמת לבנה

1.1 מטרות הפרויקט

1.1.1 חילוי פרמטרים

המערכת מחלצת מהאזורים הנבחרים בתמונה את הפרמטרים הבאים:

- מספר הנקודות
- שטח נקודה ממוצע
- צפיפות הנקודות
- מעגליות (eccentricity) ממוצעת של הנקודות – מידת המתיחה של האליפסה המזערית המכילה את הצורה. הערך נע בין אחד (מעגל מושלם) לאפס (הצורה היא קו ישר).
- מרחק מינימלי ממוצע בין נקודות
- גוון פלומה ממוצע (ללא נקודות) – פרמטר שלא היה בהישג ידם של החוקרים בשיטה הידנית

ועוד פרמטרים טכניים שמתעדים את תהליך העיבוד:

- השטח הכולל שסומן לעיבוד
- קואורדינטות הפיקסל המרכזי של אזור העיבוד

1.1.2 עמידות

צילומי התנשמות מתבצעים בשטח באמצעות מצלמה דיגיטלית. על המערכת לעמוד בפני ארבעה גורמים משתנים עיקריים, הפוגעים באחידות נתוני הקלט:

1. מטבע הדברים, תנאי התאורה אינם קבועים, ומתחלפים בין עננה לאור שמש ישיר. השוני בתנאי התאורה איננו משפיע רק על בהירות התמונה המתקבלת, אלא גם על גוון הצבע שלה.
2. המצלמה הדיגיטלית היא בעלת מצבי צילום שנבחרים שרירותית על ידי הצלם, ולכן בתנאי תאורה זהים עלולות להתקבל תמונות שונות. כמו כן, לא מובטח השימוש במצלמה יחידה.
3. ישנו שוני בין התנשמות. לעיתים מופיעים כתמי צבע כתומים על גבי הגחון, בניגוד לצפי של גחון לבן לחלוטין עם נקודות שחורות.

4. מרחק המצלמה משתנה, ומביא עצמים לכדי גדלים שונים.

1.1.3 דיוק

בבואנו למכן את פעולת החוקר, מצופה מהמערכת שתניב תוצאות שלא נופלות מיכולות העיבוד האנושי. בראש ובראשונה נדרש שאף נקודה לא תוחמץ, ולאחר מכן נדרש שהנתונים הנדלים יהיו נכונים לחלוטין. עם זאת, ישנן מספר מגבלות טכניות ידועות מראש שמונעות אמינות מוחלטת. ראשית, לנקודות שעל גחון התנשמת אין צורה מוגדרת, ולכן לא ניתן להבחין בין נקודה לבין כתם פלומה כהה או לכלוך בגודל של נקודה. שנית, לנקודות יש שוליים רכים, מה שלא מאפשר להגדיר את שטחן בצורה טריויאלית.

1.1.4 ממשק משתמש

הנחת היסוד של תכנון הממשק היא שהוא נועד להדיוטות בתחום המחשבים, ועליו להיות אינטואיטיבי ואוניברסלי ככל האפשר. לכן נעשה שימוש בשפה האנגלית, בסמלים בינלאומיים, ובעיצוב שמדריך את המשתמש.

הממשק נועד להנגיש את התהליכים הבאים:

- סימון אזור לעיבוד – לרבות ריבוע בגודל 6 על 4 ס"מ, כפי שדרשו החוקרים.
- עריכת נתונים גולמיים – הוספה, הסרה וסקירה פרטנית של נקודות.
- הצגת גוון פלומה ממוצע – ללא הממשק יש צורך בתוכנה ייעודית.

פרק 2: רקע טכני

בכדי להבין את פעולת המערכת לעומק, כפי שיפורט בפרקים הבאים, נדרש רקע טכני. בפרק זה נתאר את השיטות ששימשו בפרויקט.

2.1 קורלציה

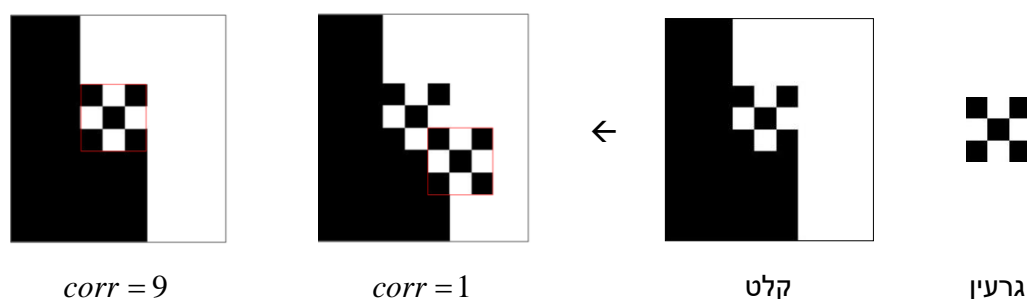
קורלציה (מתאם) בודקת את מידת ההתאמה בין משתנה תלוי לבין משתנה בלתי תלוי. אנו נדון במקרה של קורלציה בין שני מערכי פיקסלים דו מימדיים (תמונות).

בכל חישוב קורלציה, "מניחים" את מרכז גרעין הקורלציה במיקום כלשהו על פני הקלט. לאחר מכן מכפילים זה בזה את ערכי הפיקסלים החופפים של שני המערכים, וסוכמים את המכפלות לקבלת הקורלציה.

ננסח את השיטה באופן מתמטי. הקלט הוא מערך I . הגרעין B הוא מערך מגודל $M \times N$. אם נניח את מרכזו של B ב- $I[p, q]$, האזור החופף מתוך I יסומן A_{pq} . לפיכך, הקורלציה תהיה:

$$\text{corr}(I, B)[p, q] = \sum_{m=1}^M \sum_{n=1}^N A_{pq}[m, n] \cdot B[m, n]$$

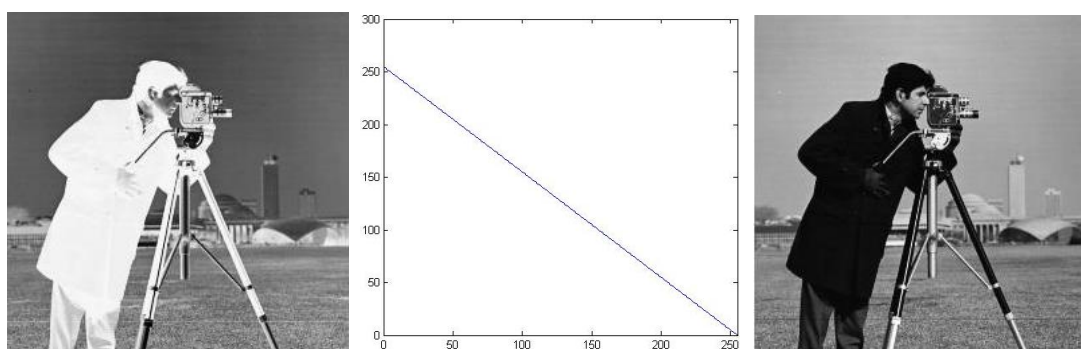
בתמונות להלן ניתן לראות תוצאת קורלציה בשני מיקומים. באחד ישנה חפיפה עם דפוס זהה לגרעין הקורלציה, ובאחד אין כזו – ועובדה זו מתבטאת בערך הקורלציה. לפיקסלים יש את הערכים ± 1 .



איור 2 – דוגמה לקורלציה

LUT 2.2

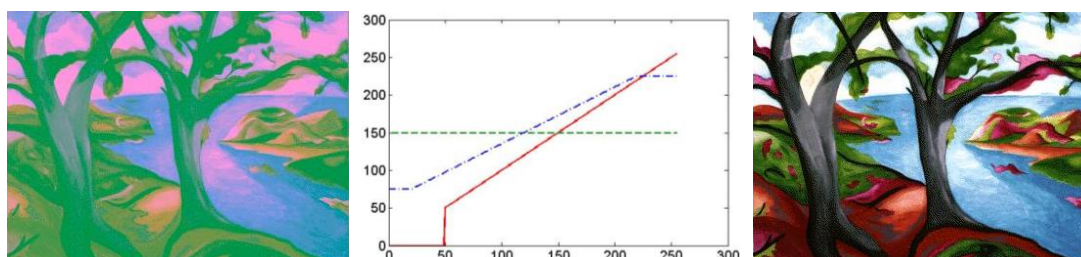
ראשי התיבות LUT משמעותם Lookup Table, או בעברית: טבלת התאמה. טבלת התאמה היא למעשה פונקציה $y(x)$ במובן הכללי ביותר. בתחום עיבוד התמונה, עבור תמונות ספרתיות אפורות בעומק 8 ביט, טבלת ההתאמה היא פונקציה בדידה $y(x): \{0, \dots, 255\} \rightarrow \{0, \dots, 255\}$. מטרתה היא להתאים לכל רמה בתמונה רמה חדשה, וכך לשנות אותה. להלן דוגמאות:



תוצאת ההתאמה

LUT

תמונה מקורית



איור 3 – דוגמאות לפעולות LUT

בדוגמה העליונה מופעלת טבלת התאמה שמהפכת את ערכי התמונה. בדוגמה התחתונה מוצגת תמונה צבעונית. כל אחד מערוצי הצבע, שמהווה לבדו תמונת אפור, מותאם דרך LUT שונה.

2.3 הרחבה וכרסום

הרחבה וכרסום הן פעולות מורפולוגיות, כלומר פעולות שמתייחסות למאפיינים גיאומטריים של תמונה. שתי הפעולות מופעלות על תמונות שחור-לבן בלבד, ומתבססות על גורם שנקרא רכיב מבני (structuring element) – גם הוא תמונת שחור-לבן.

בהגדרות הבאות נתייחס לפיקסלים הלבנים של התמונה ושל הרכיב המבני בתור "צורות" שנמצאות על רקע שחור.

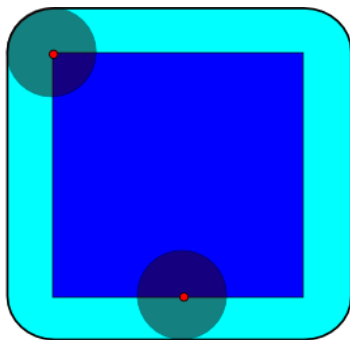
בפעולת הכרסום על צורה A , ממוקם מרכז צורת הרכיב המבני B בכל פיקסל $p \in A$ כך שהוא מסומן B_p . אם B_p לא מוכל כולו ב- A אז p מוחסר מ- A , אחרת הוא נשאר. בניסוח מתמטי:

$$A \oplus B = \{p \in A \mid B_p \subseteq A\}$$

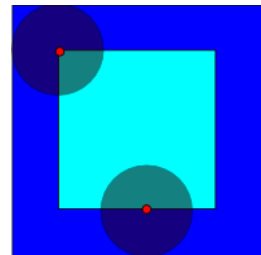
בפעולת ההרחבה על צורה A , ממוקם מרכז צורת הרכיב המבני B בכל פיקסל $p \in A$ כך שהוא מסומן B_p . כל הפיקסלים של B_p מתווספים ל- A . בניסוח מתמטי:

$$A \oplus B = \{q \in B_p \mid p \in A\}$$

בתמונה להלן ניתן לראות הדגמה של הפעולות לעיל.



דיסקה אפורה מרחיבה ריבוע כהה לבהיר



דיסקה אפורה שוחקת ריבוע כהה לבהיר

איור 4 – דוגמאות להרחבה וכרסום

הפעלת כרסום ואז הרחבה נקראת פתיחה. סדר הפעלה הפוך נקרא סגירה. נשים לב, לפי הדוגמה, שפתיחה וסגירה לא בהכרח מחזירות את הצורה ההתחלתית.

2.3.1 הרחבה וכרסום בגווני אפור

נדון במקרה הפרטי של רכיב מבני שטוח וסימטרי, כלומר כזה שערכיו הם או אפס או $-\infty$ והוא סימטרי סביב מרכזו.

בפעולת הכרסום על תמונה A , ממוקם מרכז הרכיב המבני B בכל פיקסל $p \in A$ כך שהוא מסומן B_p . ערכו החדש של p הוא הערך המינימלי של A

באזור החפיפה שלו עם B_p . בניסוח מתמטי: $(A \oplus B)(p) = \min_{q \in B} [A(p+q)]$

פעולת הכרסום מעלימה פרטים בהירים קטנים ומרחיבה כהים.

בפעולת ההרחבה על תמונה A , ממוקם מרכז הרכיב המבני B בכל פיקסל $p \in A$ כך שהוא מסומן B_p . ערכו החדש של p הוא הערך המירבי של A באזור

החפיפה שלו עם B_p . בניסוח מתמטי: $(A \oplus B)(p) = \max_{q \in B} [A(p+q)]$

פעולת ההרחבה מעלימה פרטים כהים קטנים ומרחיבה בהירים.



הרחבה עם רכיב מעגלי



תמונה מקורית



כרסום עם רכיב מעגלי

איור 5 – דוגמאות להרחבה וכרסום בגווי אפור

נשים לב שאם A תמונת שחור-לבן, הפעולות הנ"ל שקולות לאלה שנידונו קודם. כמו כן, באופן דומה לנידון קודם, פעולת כרסום ואז ההרחבה נקראת פתיחה, וסדר פעולות הפוך נקרא סגירה.

2.4 אלגוריתם k-means

אלגוריתם k-means משמש לסיווג נתונים בעלי מספר כלשהו של מימדים ל- k קבוצות. נתונים כאלה משולים לאיברים במרחב רב מימדי. האלגוריתם, שהינו איטרטיבי, מביא למינימום מקומי את הסכום, על פני כל הקבוצות, של הסכום התוך-קבוצתי של מרחקי איבר-למרכז-קבוצה.

(1) k מרכזים \rightarrow תנאי התחלה

(2) סווג כל איבר למרכז הקרוב ביותר

(3) חשב מרכז מסה חדש לכל קבוצה

(4) לכל איבר במרחב

(4.1) שנה את הסיווג שלו לקבוצה אחרת

(4.2) חשב מרכזים מחדש

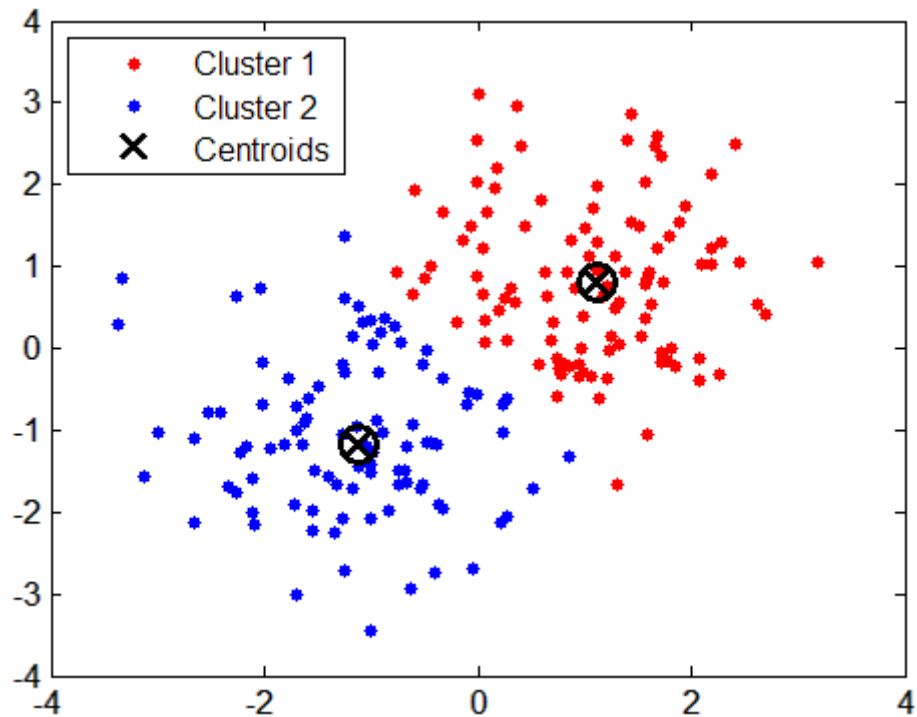
(4.3) אם סכום המרחקים איבר-מרכז של כל הקבוצות לא קטן

(4.3.1) בטל את פעולות (4.1) ו-(4.2)

(4.3.2) חזור ל-(4.1) עם קבוצה שלא נוסתה

(5) חזור ל-(2) כל עוד המרכזים משתנים מעבר ל- ϵ

להלן דוגמה של סיווג אברים דו מימדיים:



פרק 3: תיאור כללי

המערכת מורכבת ממספר תהליכי עיבוד המתווכים בין קלט של תמונה גולמית לבין פלט של מאגר נתונים. להלן תיאור שטחי של תהליכים אלה, שיפורט לעומק בפרק הבא.

3.1 זיהוי כרטיס כיול

בתמונות שמוזנות למערכת, לצד התנשמת אמור להופיע בשלמותו כרטיס כיול של חברת WhiBal, ולו גודל ידוע מראש. הכרטיס מתאפיין בגווני אפור ובצורת שמש הנמצאת באחת מפינותיו.

המערכת מזהה את הכרטיס בזכות העובדה שמראהו ידוע מראש. מתוך מחשבה על יעילות, האלגוריתם תוכנן להתבצע בשלבים, כשכל שלב מזהה מאפיין בודד של מראה הכרטיס.

בשלב הראשון, המערכת מזהה את מרכז דפוס השמש. המערכת מניחה כי הדפוס לא מופיע בשום מקום בתמונה מלבד כרטיס הכיול. בשלב השני, המערכת מוצאת את הזווית והאורך של הקו המחבר בין מרכז השמש לבין קצה הצלע הארוכה של הכרטיס (משמאל למטה לשמאל למעלה בתמונה). שני נתונים אלה הם הנחוצים למערכת לצורך ביצוע השלב הבא.



איור 6 – כרטיס הכיול איתו עובדת המערכת

3.2 כיול

3.2.1 קנה מידה

כאמור בפרק המבוא, בין פלטי המערכת נמנים גדלים כמו שטח ומרחק. קיים צורך להמיר את יחידות המידה של המחשב - הפיקסלים, ליחידות מידה של בני אדם – ס"מ, סמ"ר וכדומה.

יחס ההמרה פיקסל-לסנטימטר מחושב לאחר שלב זיהוי כרטיס הכיול, במהלכו נמצא גודל הכרטיס בפיקסלים. גודל הכרטיס בסמ"ר ידוע מראש. ההשוואה בין גדלים אלה מניבה את קנה המידה. בסוף התהליך, מופק היחס האמור. יצוין שהמערכת מניחה שכרטיס הכיול הינו פחות-או-יותר באותו מרחק מהמצלמה כמו הנקודות, שכן אחרת קנה המידה איננו רלוונטי לנקודות.

3.2.2 צבעים

אחד מפלטי המערכת, כפי שהוזכר בפרק המבוא, הוא גוון פלומה ממוצע. עוד הוזכר שתנאי הצילום משתנים בין תמונה לתמונה, ומשפיעים על הגוונים הנראים בה. על מנת לבטל השפעה זו, ולדאוג לכך שגוון הפלומה של תנשמת מסוימת יהיה זהה בכל תנאי התאורה, יש לכייל את הצבעים בתמונות. בנוסף לכך, הכיול תורם גם לזיהוי הנקודות, שמושפע מבהירותן. כיול הצבעים מתבצע בעזרת התאמה בין צבעי כרטיס הכיול הידועים מראש לבין צבעי כרטיס הכיול כפי שהם נראים בפועל בתמונה. התמונה המכילת היא זו שתעובד ע"י המערכת מנקודה זו ואילך.

3.3 זיהוי נקודות

הנקודות הפזורות על גחונה של התנשמת מתאפיינות בפרטים הבאים:

- בהירות נמוכה
- צורה מעגלית או אליפטית
- מגבלת גודל

המערכת משתמשת במאפיינים אלה בתהליך הזיהוי. תהליך זה אינו מתמודד עם תמונת התנשמת השלמה, אלא מקבל אזור על הגחון כקלט שמסומן ע"י

המשתמש. האלגוריתם מזהה את הנקודות, ומפיק תמונת שחור-לבן המתארת את מיקומן.

3.4 הוספת נקודה נבחרת

לעיתים קיימות נקודות שמנגנון הזיהוי לא מזהה אותן. כפי שהוזכר בחלק 1.1.4, למשתמש ניתנת האפשרות להוסיף נקודות כאלה. בעזרת הממשק, המשתמש לוחץ על הנקודה וכך מספק למערכת מידע חלקי על מיקומה.

מנגנון זיהוי הנקודה הנבחרת הוא גרסה מתירנית (מונח שיוסבר בחלק 4.4) של מנגנון זיהוי הנקודות, הלוקחת בחשבון את המיקום שנמסר ע"י המשתמש. המנגנון מזהה את הנקודה, ומפיק תמונת שחור-לבן המתארת את מיקומה.

3.5 הפקת נתונים

משזוהו הנקודות, הנתונים לגביהן, המוזכרים בפרק המבוא, מחולצים על ידי חישובים מתמטיים שתוצאותיהם מומרות ליחידות מתאימות. ההמרה מתבצעת בעזרת קנה המידה שנמצא בשלב הכיול. גוון הפלומה הממוצע מחושב ע"י התייחסות בלעדית לאזורים שלא התגלו בהם נקודות.

פרק 4: מימוש

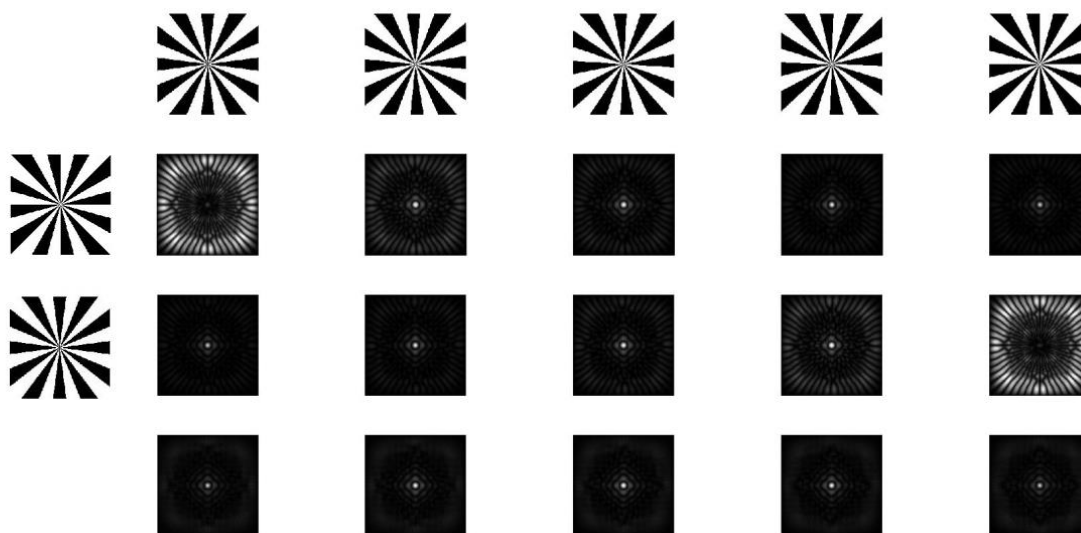
בפרק הזה נסביר באופן מפורט את תהליכי העיבוד שנמנו בפרק הקודם, ונציג את האלגוריתמים שעומדים מאחוריהם.

4.1 זיהוי כרטיס כיול

כפי שהוזכר, זיהוי הכרטיס מחולק לשני שלבים. כל שלב מחפש מאפיין אחר של הכרטיס בעזרת קורלציה. שיטה זו הוסברה בפרק הרקע הטכני. כעת נעסוק בשיקולי התכנון של גרעיני הקורלציה בשני השלבים.

4.1.1 מיקום

כרטיס הכיול יכול להופיע בכל זווית, אך דפוס השמש, הודות למחזוריותו המעגלית, נראה זהה בכל 30 מעלות. לפיכך, ניתן לזהות את השמש לפי קורלציה עם שני גרעיני קורלציה בצורת שמש, המסובבים ב-15 מעלות זה ביחס לזה. נסביר מדוע באיור הבא:



איור 7 – טבלת קורלציות. אזורים בהירים בכל תמונה מייצגים ערכים גבוהים באופן יחסי לתמונה הבודדת בלבד.

בטור השמאלי של האיור לעיל מוצגים שני הגרעינים, ובשורה העליונה הם מוצגים שוב בקצוות, כשביניהם דפוסי שמש בזוויות ביניים. במרכז מוצגות קורלציות בהצלבה מתאימה, בערך המוחלט.

ניתן לראות שמתקבל מקסימום חזק במרכז כל קורלציה, בכל הזוויות פרט למקרה בו הקלט והגרעין מסובבים ב-15 מעלות זה לזה. לכן לא די בגרעין אחד, אלא צריך גרעין נוסף כמתואר. כך, כשאחד מגיע לשפל, השני מגיע לשיא.

בשורה התחתונה של האיור מוצג סכום שתי השורות שמעליה. ניתן לראות שסכימה שכזו מבטיחה שיא במרכז לכל זווית שהיא.

גרעיני הקורלציה הם תמונות בינאריות שערכיהן הם $\pm a$ כאשר a נבחר שרירותית להיות 0.5. ישנן 2 סיבות לכך:

1. כאשר יש פיקסל בעל ערך 0 בגרעין, הפיקסל הוא בבחינת don't care,

שכן כל פיקסל שחופף איתו מניב את אותה מכפלה (אפס) ולא משפיע

על ערך הקורלציה. לכן אנחנו לא מעוניינים בפילוג כמו $\{0,1\}$.

2. גרעין שערכיו הפוכים (שחור-לבן \leftarrow לבן-שחור) מניב קורלציה שונה, שכן

המכפלות משתנות. אם משתמשים בערכים $\pm a$ אז מקבלים קורלציה

זוהי, שהפוכה רק בסימן. לכן, אם המערכת תעבוד עם הערך המוחלט

של הקורלציה – גרעין יחיד יבצע פעולה של שניים.

נבהיר, שבגלל סיבה 2 המערכת משתמשת ב-2 גרעינים במקום ב-4. על מנת

שהשמש בתמונה תהיה דומה ככל האפשר לגרעין הקורלציה, התמונה מומרת

לגווני אפור, וערכיה משויכים לתחום $[-0.5, 0.5]$.

גודלם של גרעיני הקורלציה, יחסית לגודל השמש בתמונה, הוא סוגיה בפני

עצמה. בקביעתו יש לקחת בחשבון את המקרים הבאים:

גודל גרעין	יתרונות	חסרונות
מדויק	יחס אופטימלי בין הצלחה לזמן חישוב	בלתי ניתן לחיזוי, כי בכל תמונה הכרטיס מצולם ממרחק אחר
קטן	<ul style="list-style-type: none"> זמן חישוב קצר טוב בתיאוריה, מטעמי סימטריה 	<ul style="list-style-type: none"> רגיש להתחזזות (aliasing) במרכז השמש, במיוחד תחת מיקוד ירוד
גדול	<ul style="list-style-type: none"> לא רגיש לקיפול טוב מטעמי סימטריה 	<ul style="list-style-type: none"> זמן חישוב ארוך ערך הקורלציה מושפע מפיסקלים לא רלוונטיים רבים, ולכן המקסימום שמחפשים מאבד מחדותו

לכן הבחירה הטובה ביותר היא גודל מדויק ככל האפשר, עם נטיה להגדלה. בפועל, הגודל נקבע על ידי הערכה גסה. נמצא כי שטח השמש הוא כ-3.1% משטח הכרטיס, ואת שטח הכרטיס באחוזים מהתמונה המשתמש יכול להזין, כשברירת המחדל האמפירית היא 3.6%.

לסיכום, האלגוריתם הוא כדלקמן:

- (1) המר את התמונה לגווני אפור
- (2) שייך את ערכי התמונה לתחום $[-0.5, 0.5]$
- (3) צור גרעין שמש בערכים ± 0.5 ששטחו כ-0.11% מהתמונה
- (4) צור גרעין זהה ומסובב ב- 15°
- (5) חשב קורלציות בין התמונה לגרעינים
- (6) סכום ערכים מוחלטים של הקורלציות
- (7) מצא פיקסל בעל ערך מקסימלי בתמונת הסכום

הפיקסל הנ"ל הוא מיקום מרכז השמש של כרטיס הכיול.

4.1.2 גודל זווית

שלב זה מקבל מהשלב הקודם את מיקום מרכז השמש של כרטיס הכיול. מראה כרטיס הכיול ידוע מראש, ובפרט ידועה שורת הפיקסלים היוצאת ממרכז השמש ומקבילה לצלע הארוכה של הכרטיס, כמתואר באיור.

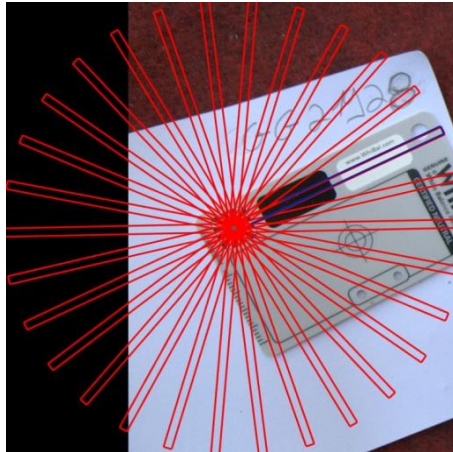


איור 8 - חלקו העליון של כרטיס הכיול

כמו בשלב הקודם, גם בשלב הזה נשתמש בקורלציה ונחפש את ערכה המירבי. גרעין הקורלציה יהיה פס הפיסקלים המסומן בתמונה הנ"ל, והינו הרצף החד מימדי הבא (באחוזים ממרכז השמש ועד קצה כרטיס הכיול):

אפור	לבן	אפור	שחור	אפור	פנים השמש
83.8%-100%	52%-83.8%	46.7%-52%	13.7%-46.7%	8.5%-13.7%	0%-8.5%

בניגוד לשלב הקודם, גרעין הקורלציה לא ינוע ברחבי התמונה. קצה אחד של הגרעין תמיד יהיה במרכז השמש, והגרעין יטה בשלל זוויות, כמתואר באיור להלן.



איור 9 – זוויות הקורלציה. הזווית הנבחרת מסומנת בכחול. חלקה השמאלי של הסריקה חורג מגבולות התמונה ולכן היא מרופדת באפסים.

באופן שרירותי, נקבע שהמרחב יחולק אחידות ל-30 זוויות בדיקה התחלתיות.

אך כידוע, כדי לייצר גרעין יש לקבוע את אורכו, והרי שהאורך לא ידוע לפני שיודעים את אורך הכרטיס. לכן מתבצעת קורלציה עם 20 גרעינים באורכים שונים. האורכים מפוזרים אחידות בין שלושה לבין 30 רדיוסי שמש משוערים (ראו מהו שטח שמש משוער בשלב הקודם).

לאחר שנסקו 30 זוויות עם 20 האורכים, נבחרים האורך והזווית שהניבו קורלציה מירבית. כעת נבחרות עוד 5 זוויות קרובות ו-5 אורכים קרובים, מתבצעות קורלציות, ונבחר השילוב אורך-זווית הטוב ביותר.

תיאורטית, שיטה זו מבטיחה שגיאת זווית מקסימלית של 1.2° בלבד, ושגיאת אורך של 27% מרדיוס שמש משוער.

גרעיני הקורלציה נבחרו להיות בעלי ערכים ± 0.5 עבור שחור ולבן, וערך 0 עבור אפור. התמונה מומרת לגווני אפור, וערכיה משויכים לתחום $[-0.5, 0.5]$. הסיבה להחלטות אלה הוזכרה בחלק 4.1.1.

באיור להלן ניתן לראות את הערכים של גרעינים בשלושה אורכים שונים.



איור 10 - שלושה גרעיני קורלציה. אחד סומן בשחור, השני שקוף עם קו מתאר אדום, והשלישי בסימון אפור. כל גרעין הוא חד מימדי, והגרף לעיל מסמן שטח רק לצורכי הבליטה ויזואלית.

ניתן לראות כי תחילתו של הגרעין, שחופפת עם פנים השמש, קיבלה את הערך אפס. הסיבה לכך היא שפנים השמש הוא תלוי זווית, ולכן בלתי צפוי, ומכאן שמתאים לסמנו בערך אפס, שמתפקד כ-don't care בחישוב הקורלציה.

לסיכום, האלגוריתם הוא כדלקמן:

(1) חשב את מיקום מרכז השמש
(2) השתמש בתמונה שעליה חושבה קורלציה בשלב (1)
(3) לכל אורך x מתוך 5 אורכים
(3.1) לכל זווית θ מתוך 30 זוויות בין 0° ל- 360°
(3.1.1) צור גרעין באורך x , מוטה בזווית θ
(3.1.2) חשב קורלציה כשקצה הגרעין במרכז השמש
(4) מצא שילוב אורך-זווית בעל קורלציה אופטימלית
(5) לכל אורך w מתוך 5 אורכים קרובים ל- x אופטימלי
(5.1) לכל זווית α מתוך 5 זוויות קרובות ל- θ אופטימלי
(5.1.1) צור גרעין באורך w , מוטה בזווית α
(5.1.2) חשב קורלציה כשקצה הגרעין במרכז השמש
(6) מצא שילוב אורך-זווית בעל קורלציה אופטימלית

4.2 כיול

בתהליך זה, המערכת משתמשת בנתונים שחולצו אודות כרטיס הכיול.

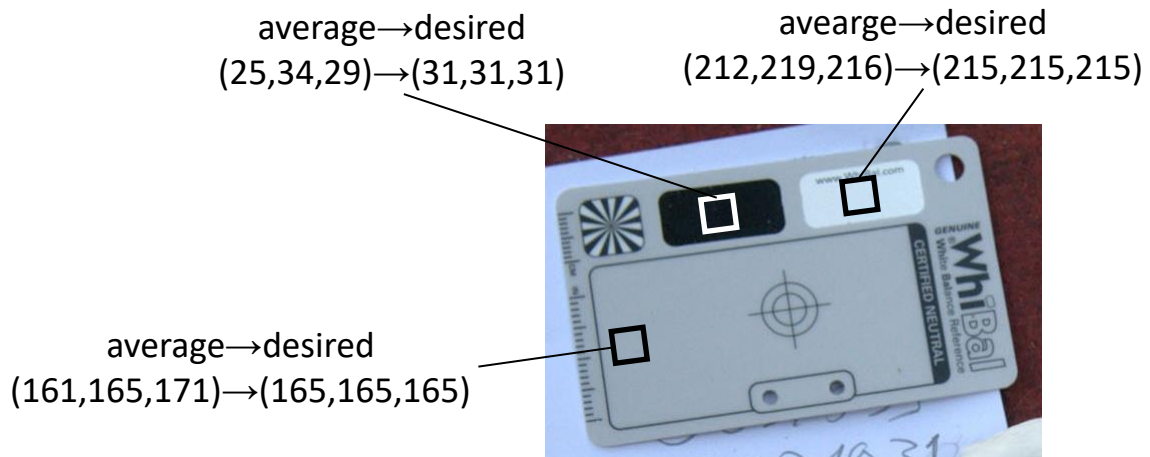
4.2.1 קנה מידה

ממדי כרטיס הכיול במציאות ידועים לנו. לאחר שלב זיהוי הכרטיס, ידוע אורכו בפיסקלים. כל שנותר על מנת למצוא את קנה המידה, דהיינו למצוא את היחס בין פיקסלים לסנטימטרים בתמונה, הוא לחשב כך:

$$\text{קנה מידה} = \frac{\text{אורך הכרטיס בפיסקלים}}{\text{אורך הכרטיס בסנטימטרים}}$$

4.2.2 צבעים

כיול הצבעים נעשה באמצעות דגימת מקומות מסוימים על כרטיס הכיול. מקומות אלה, כפי שניתן לראות באיור להלן, מייצגים את שלושת גווני הכרטיס.



איור 11 – דגימת אזורים על הכרטיס והתאמתם לערכים רצויים.

איתור מיקומי הדגימה נעשה בזכות המידע שהופק בשלב זיהוי הכרטיס. כלומר המיקומים מתוארים בעזרת מיקום השמש, אורך הכרטיס וזויתו. תיאור זה מוצג בפסאודו-קוד שבסוף הסעיף.

האזורים הם בגודל 10 על 10 פיקסלים, ודגימתם היא למעשה מיצוע ערכי ה-RGB של הפיקסלים, לקבלת פיקסל ממוצע. בסך הכל מתקבלים 3 פיקסלים ממוצעים כאלה. ל-9 ערכי ה-RGB שהתקבלו נקרא ערכים מצויים.

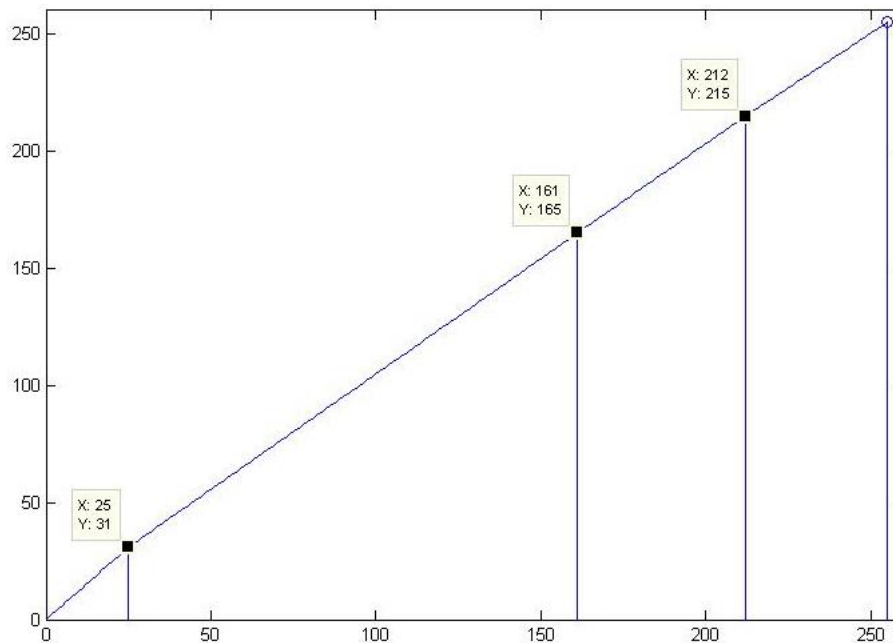
התמונה מכילות ע"י LUT שמתאים את הערכים המצויים לערכים רצויים שנבחרו תחת השיקולים הבאים:

1. לובן הכרטיס איננו האזור הבהיר ביותר בתמונה. אם נשייך אותו ללבן בהיר מאוד, האזורים שבהירים ממנו "יישרפו".
2. החלק השחור של הכרטיס איננו האזור הכהה ביותר בתמונה. אם נשייך אותו לשחור כהה מאוד, האזורים שבהים ממנו "יעלמו".

בהתחשב בזאת, נבחרו באופן שרירותי הפיקסלים הרצויים הבאים:

לבן: (215,215,215) אפור: (165,165,165) שחור: (31,31,31)

בניית ה-LUT נעשית בנפרד לכל אחד מהערוצים RGB, ע"י אינטרפולציה לינארית, כמתואר בשרטוט להלן. יש לציין כי השימוש ב-LUT מועיל במקרים של עיוות צבעים לינארי, אך לא במקרים של פעולת גאמה מעוותת.



איור 12 – LUT עבור הערוץ האדום של הדוגמה באיור הקודם

לסיכום, האלגוריתם הוא כדלקמן:

- (1) התקדם ממרכז השמש בזווית הכרטיס כ-27% מאורך הכרטיס, ודגום את הסביבה הקרובה
- (2) התקדם ממרכז השמש בזווית הכרטיס כ-62% מאורך הכרטיס, ודגום את הסביבה הקרובה
- (3) התקדם ממרכז השמש בניצב לזווית הכרטיס כ-30% מאורך הכרטיס, ודגום את הסביבה הקרובה
- (4) לכל ערוץ בתמונה
 - (4.1) בנה דטט בן 256 ערכים שמתאים את הערכים שנדגמו לערכים רצויים
 - (4.2) הפעל את ה-דטט על הערוץ

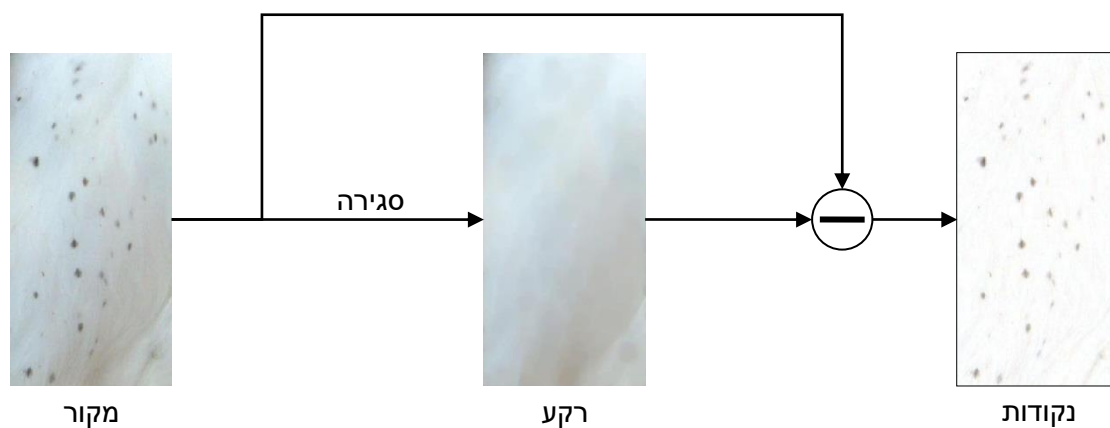
4.3 זיהוי נקודות

כאמור בפרק התיאור הכללי, זיהוי הנקודות מתבצע על אזור שסומן על ידי המשתמש. המנגנון מוחק תחילה את הרקע של הפלומה מסביב לנקודות ע"י פעולות מורפולוגיות, שהוסברו בפרק הרקע הטכני.

על מנת לבצע פעולה מורפולוגית יש צורך ברכיב מבני. ידוע ששטח משוער של נקודה הוא 0.135 סמ"ר, וכן ידוע שצורתה המשוערת היא אליפטית. לפי

הערכות גסות אלה, הרכיב המבני נקבע להיות מעגל ששטחו בפיקסלים הוא המרה של שטח הנקודה המשוער, לפי קנה המידה של התמונה.

בעזרת הרכיב המבני, מופעלת פעולת סגירה על התמונה, כלומר כל ערוץ צבע עובר הרחבה ואז כרסום בגווי אפור, כפי שתוארו בחלק 2.3.1. פעולת ההרחבה מצמצמת את האובייקטים הכהים, וכך נעלמים אובייקטים כהים בגודל הרכיב המבני ומטה – דבר שגורם לנקודות להיעלם. לאחר מכן, פעולת הכרסום מחזירה את האובייקטים שלא נעלמו לגודלם המקורי בקירוב. כלומר, מתקבלת תמונה של הרקע. לאחר מכן, תמונת הרקע מחוסרת מהתמונה המקורית וכך רק הנקודות "שורדות". תהליך זה מתואר באיור להלן.



איור 13 – סכמה של תהליך bottom hat

יש לציין שהפעולה המורפולוגית נעשית על כל ערוץ בנפרד, שמהווה לבדו תמונה בגווי אפור. יצוין גם כי הפעולה הכוללת המתוארת נקראת סינון bottom hat.

כעת, תמונת הנקודות עוברת סיווג פיקסלים אוטומטי ל-2 קבוצות, בעזרת אלגוריתם k-means שהוסבר בפרק הרקע הטכני. הקבוצות הן:

1. פיקסלי הנקודות הכהים.

2. הפיקסלים הבהירים של הפלומה.

הפיקסלים מוזנים לאלגוריתם כקואורדינטות במרחב RGB, התלת מימדי. תנאי ההתחלה של האלגוריתם נקבעו אמפירית בהתאם לתמונות שהניב אלגוריתם k-means:

1. פיקסל נקודה טיפואי בעל $RGB=200$ כ"א.

2. פיקסל פלומה טיפואי בעל $RGB=255$ כ"א.

יש לציין כי אין המערכת רגישה במיוחד לתנאי ההתחלה, ועיקר תפקידם הוא למנוע התחלפות בסימוני הקבוצות.



איור 14 – סכמה של סיווג k-means

מפת הפיקסלים שתויגו כפיקסלי נקודות מופקת כתמונה בינארית. היא מחולקת לאובייקטים (connected components) ומוסרים ממנה אובייקטים חריגים בגודלם: קטנים או גדולים בהרבה מגודל נקודה משוער. כמו כן, הוסרו אובייקטים שמייצגים אזורים בהירים בצורה חריגה.

לסיכום, האלגוריתם הוא כדלקמן:

- (1) צור רכיב מבני עגול בשטח 0.135 סמ"ר
- (2) בצע בעזרת הרכיב המבני סינון bottom hat על התמונה
- (3) סווג את תוצאת הסינון ל-2 קבוצות באלגוריתם k-means
- (4) צור מסכת נקודות מהפיקסלים שסווגו כפיקסלי נקודות
- (5) הוצא מהמסכה הנ"ל אובייקטים קטנים מדי, גדולים מדי, או בהירים מדי

4.4 הוספת נקודה נבחרת

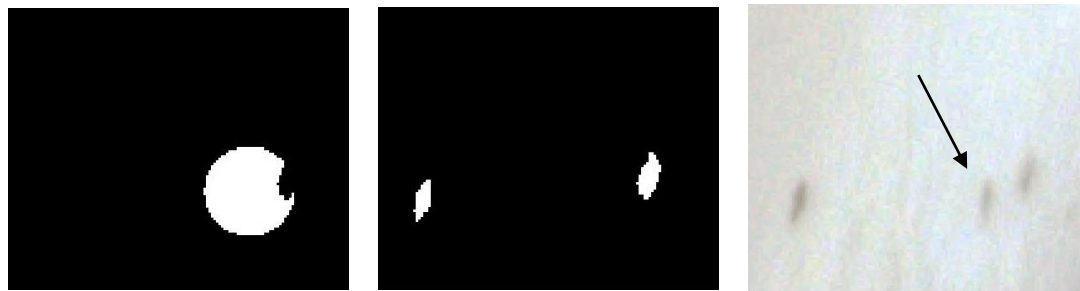
בתחילת התהליך, המשתמש בוחר את הנקודה. הוא עושה זאת ע"י סימון של פיקסל יחיד על גביה. כלומר, למערכת מוזן מיקום אחד הפיקסלים של הנקודה.

לאחר מכן, מופעל תהליך זיהוי הנקודות כמתואר בחלק 4.3, אלא שהוא מתייחס רק לסביבה הקרובה של הסימון, ברדיוס של 2 רדיוסי נקודה משוערת. האלגוריתם מבצע סינון bottom hat באופן זהה למתואר קודם. סיווג הפיקסלים באלגוריתם k-means מתבצע באופן מעט שונה:

- תנאי ההתחלה מקומיים – נקבעו להיות הפיקסל הכי בהיר והכי כהה בסביבה הנ"ל

- התעלמות מנקודות קיימות – אין התייחסות לפיקסלים שכבר זוהו כנקודות בשלב 3.3, ונכללים בסביבה הקרובה. הסיבה לכך היא שלעיתים הנקודה המבוקשת היא שונה למדי מהנקודות הסמוכות אליה, ולכן עלולה להיות מסווגת כפלומה אם תסווג יחד עם נקודות אחרות.

באיור להלן ניתן לראות את המחשת הסעיף השני:



סביבת ההתייחסות של k-means

שתי נקודות כבר מזוהות

תמונה מקורית

איור 15 – התעלמות מנקודות מזוהות במהלך הוספת נקודה נבחרת

באיור מופעל זיהוי נקודה נבחרת על הנקודה האמצעית, כמסומן בחץ.

תוצאת k-means עלולה להכיל מספר אובייקטים, לכן נבחר רק האובייקט שיש לו פיקסל משותף עם בחירת המשתמש. לאחר מכן האובייקט מוסר אם הוא גדול בהרבה מגודל נקודה משוער. אין הסרה של אובייקט בהיר מדי או קטן מדי.

לסיכום, האלגוריתם הוא כדלקמן:

(1) קלוט מיקום p מהמשתמש
(2) צור רכיב מבני עגול בשטח 0.135 סמ"ר
(3) בצע בעזרת הרכיב המבני סינון bottom hat על התמונה

- (4) התייחס רק לסביבה ברדיוס 0.415 ס"מ מ- p והתעלם מנקודות מזהות שנמצאות בה
- (5) סווג את אזור ההתייחסות ל-2 קבוצות באלגוריתם k -means
- (6) צור מסכת נקודות מהפיקסלים שסווגו כפיקסלי נקודות
- (7) השאר במסכה רק את האובייקט שמכיל את p
- (8) הוצא מהמסכה הנ"ל אובייקטים גדולים מדי

4.5 הפקת נתונים

מסכת הנקודות, שהינה תמונה בינארית, מחולקת לעצמים בעזרת פונקציית `bwconncomp` של MATLAB. כל עצם הוא קבוצה קשירה של פיקסלים בעלי הערך '1' ונחשב לנקודה אחת על גחון התנשמת. בעזרת פונקציית `regionprops` של MATLAB, מחושב המידע הבאים עבור כל עצם:

- שטח
- מעגליות (eccentricity)
- מרכז מסה

לאחר מכן, מופקים הנתונים הרצויים, שנמנו בפרק המבוא, בעזרת החישובים להלן (יפ"ס – יחס פיקסל-לסנטימטר).

$$\text{שטח נקודה ממוצע} = \frac{\text{סכום השטחים}}{\text{מספר הנקודות} \times \text{יפ"ס}^2} \quad [\text{סמ"ר}]$$

$$\text{צפיפות הנקודות} = \frac{\text{מספר הנקודות} \times \text{יפ"ס}^2}{\text{שטח האזור שעובד}} \quad [\text{נקודות לסמ"ר}]$$

$$\text{מעגליות ממוצעת} = \frac{\text{סכום המעגליות עפ"נ כל הנקודות}}{\text{מספר הנקודות}} \quad [\text{ללא יחידות}]$$

$$\text{מרחק מינימלי ממוצע בין נקודות} = \frac{\text{סכום המרחקים המינימליים בין מרכזי מסות}}{\text{מספר הנקודות} \times \text{יפ"ס}} \quad [\text{ס"מ}]$$

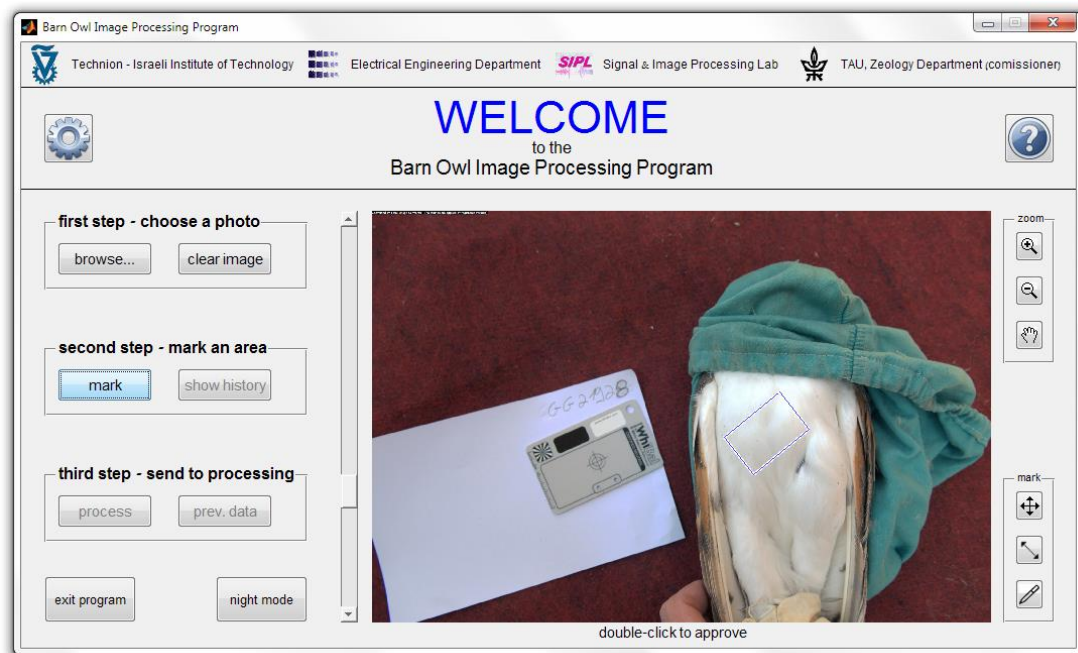
יחידות המידה ניתנות לשינוי, לפי בחירת המשתמש.

נתון נוסף שמופק הוא גוון הפלומה הממוצע. זהו למעשה פיקסל הפלומה הממוצע, כלומר שלישיית RGB. הוא מחושב ע"י מיצוע ערכי הפיקסלים שזוהו כפלומה.

פרק 5: ממשק משתמש

התהליכים שהוסברו בפרקים הקודמים מופעלים ע"י המשתמש בעזרת ממשק משתמש.

עם הפעלת המערכת, עולה המסך הראשי של הממשק. עיקר תכליתו היא לטעון תמונה ולסמן אזור לעיבוד על גחון התנשמת.



איור 16 – המסך הראשי במהלך סימון בצורת ריבוע מסתובב בגודל קבוע

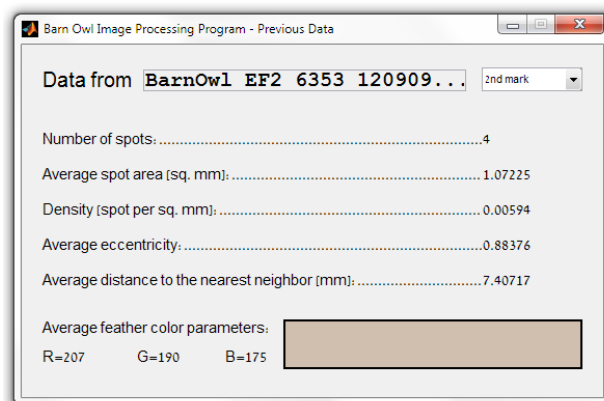
למסך מספר כפתורים עיקריים:

1. "browse..." – בחירת תמונת תנשמת. עם אישור הבחירה, התמונה עוברת כיול במהלכו מוצגת הודעת המתנה, ולאחר מכן מוצגת במסך הראשי יחד עם פקדי תקריב מתאימים.
2. "clear image" – מאפס את הממשק לשלב ההתחלתי, בו אין תמונה שטעונה למערכת.
3. "mark" – מציג פקדי סימון בפינה הימנית התחתונה. הפקדים מציעים שלוש אפשרויות סימון:

a. ריבוע בעל גודל קבוע, הניתן להזזה וסיבוב

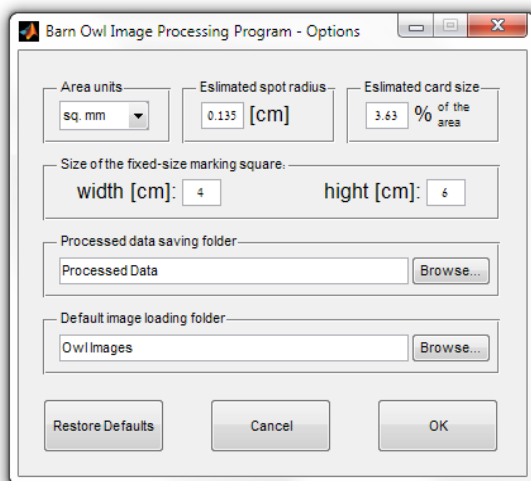
b. ריבוע בעל גודל משתנה, הניתן להזזה

- c. צורה חופשית, הניתנת להזזה לאחר ציורה
4. "show history" – מחליף את תמונת התנשמת בממשק, בתמונה עליה צבועים אזורי הסימון שעובדו קודם לכן, בצבע שקוף למחצה.
5. "process" – אישור האזור שסומן ושליחתו לעיבוד.
6. "prev. data" – הצגת תוצאות עיבודים קודמים של התמונה הנוכחית, מאז טעינת התמונה. התוצאות מוצגות במסך משלהן, כמו זה שמוצג בתמונה להלן.

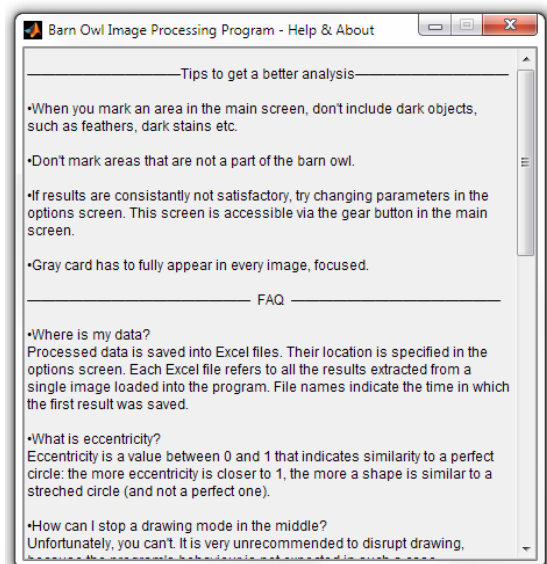


איור 17 – מסך תוצאות קודמות. ממחיש את גוון הפלומה הממוצע

המסך הראשי מאפשר גישה למסכי העזרה והאפשרויות.



איור 19 – מסך האפשרויות

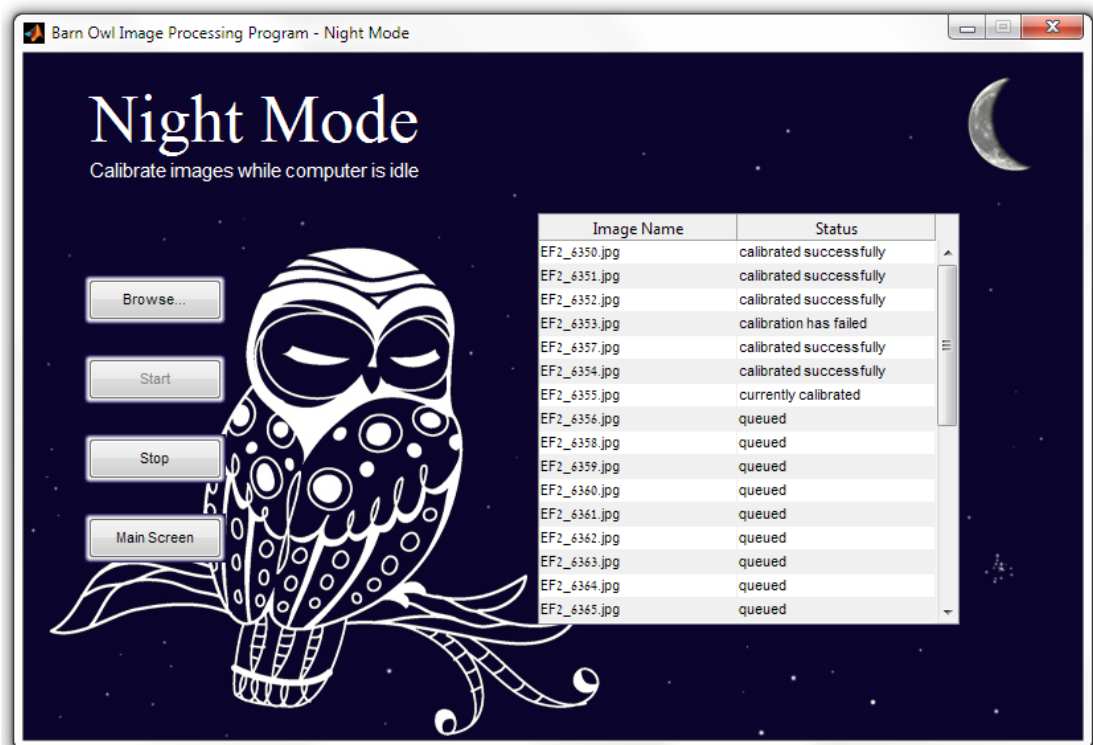


איור 18 – מסך העזרה

במסך העזרה ניתנות עצות לזיהוי נקודות טוב יותר, מענה לשאלות נפוצות, פתרון תקלות, ומידע כללי אודות התוכנה. במסך האפשרויות ניתן לשנות את הפרמטרים הבאים:

- יחידות מידה רצויות לשמירת נתונים
 - גודל משוער של נקודה (בס"מ)
 - גודל משוער של כרטיס כיול (באחוזים מהתמונה)
 - ממדים רצויים של מלבן סימון קבוע (בס"מ)
 - תיקייה לשמירת הנתונים
 - תיקיית ברירת מחדל לטעינת תמונות
 - ערכי RGB משוערים של נקודה ושל פלומה.
- תמיד ניתן לשחזר את הפרמטרים לכדי ברירת המחדל.

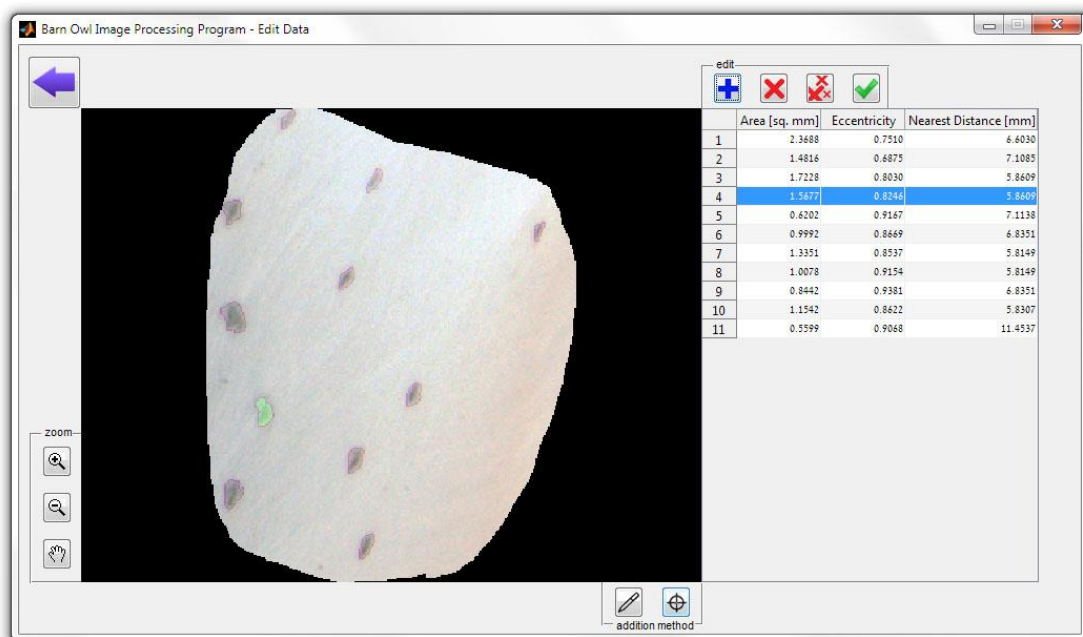
כמו כן, דרך המסך הראשי ניתן לגשת ל"מצב לילה". זהו מנגנון שמאפשר לכייל תמונות, פעולה הלוקחת זמן רב עבור רזולוציה גבוהה, באופן אוטומטי בזו אחר זו, בשעות נוחות.



איור 20 – מצב לילה בזמן הכיול, מציגה מגוון הודעות על מצב פעולתה.

לאחר כל כיול, נשמר קובץ משתנים לצד התמונה, ובו קנה המידה שלה וגרסה מכוילת שלה. כאשר היא תיטען למסך הראשי בבוא הזמן ע"י המשתמש, הממשק יזהה את המשתנים שנשמרו לצידה. הוא לא יבצע כיול, וישתמש בהם במקום.

במסך הראשי, לאחר סימון אזור על גחון התנשמת, לחיצה על כפתור process מפעילה את מנגנון זיהוי הנקודות ומעבירה את המשתמש למסך עריכת הנתונים. במסך זה, המשתמש מוזמן לסקור את הנקודות שזוהו ואת פרטיהן, ולמחוק אותן או להוסיף חדשות במקרה הצורך.




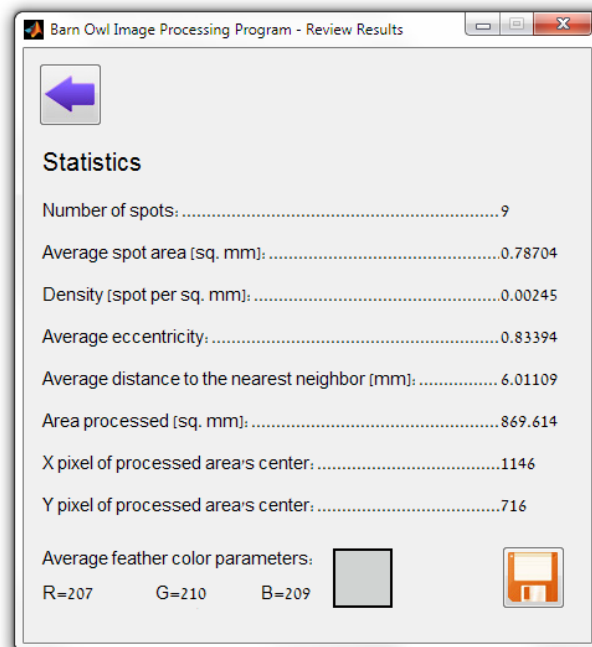
איור 21 – מסך עריכת הנתונים. הנקודות מוקפות בסגול, והנקודה הנבחרת בטבלה מסומנת בירוק.

הממשק מציג תמונה גזורה של אזור העיבוד, בה מוקפות הנקודות שזוהו, ולצידה פקדי תקריב. מצידה השני מופיעה טבלה, ובה נתוני הנקודות ביחידות מידה שנבחרו במסך האפשרויות. לחיצה על שורה בטבלה גורמת להדגשת הנקודה הרלוונטית בתמונה.

עריכת הנתונים מתאפשרת בעזרת הפעולות הבאות:

1. מחיקת נקודה בודדת – ע"י בחירתה בטבלה ולחיצה על ✖
2. מחיקת נקודות מרובות – לחיצה על כפתור ✖ מציג פקדים המאפשרים למחוק נקודות ע"י:

- a. לחיצה עליהן
- b. סימון אזור למחיקה
3. הוספת נקודה – לחיצה על כפתור + מציג פקדים המאפשרים הוספת נקודה ע"י:
- a. ציור בצורה חופשית
- b. לחיצה על אזור בתמונה, שמפעילה מנגנון הוספת נקודה (שמתואר בחלקים 3.4 ו-4.4)
- לאחר שאושרו הנקודות ע"י הכפתור , נדלים נתונים ממסכת הנקודות, והמשתמש מועבר למסך הצגת התוצאות.



איור 22 – מסך הצגת התוצאות

המשתמש יכול לסקור את הנתונים שהופקו, המוצגים לו ביחידות המידה שנבחרו במסך האפשרויות, ולשמור אותם. השמירה מתבצעת לתוך קובץ גליון נתונים של תוכנת Excel, תחת שם התמונה בתוספת חותמת זמן.

5.1 פתרונות מיוחדים

במהלך בניית הממשק נעשה שימוש חכם בקוד על מנת לממש כלים שלא מובנים בכלי GUIDE של MATLAB.

מלבן הסימון המסתובב הוא אובייקט מסוג `imroi` שנוצר ע"י פונק' `impoly` שהוזנו לתוכה 4 מיקומים (כל אחד הוא קודקוד). MATLAB מקנה לאובייקטים כאלה התנהגות אינטראקטיבית, הכוללת הזזה, שינוי צורה ועוד – אך לא סיבוב.

תכונת הסיבוב ממומשת כך: בזמן שהמלבן מונח על הצירים שומרים את המצביע (`handle`) אליו. לאחר מכן, מופעלת פונקציית `wait` על המצביע, ומופיע בממשק פס גרירה (ניתן לראותו בתמונת המסך הראשי הנ"ל). בכל פעם שנבחר ערך חדש מפס הגרירה, מעודכנים קודקודי המלבן בעזרת `setPosition` לפי המשוואות הבאות:

$$x_{new} = x_{center} + (x_{old} - x_{center}) \cdot \cos(\theta_{new} - \theta_{old}) - (y_{old} - y_{center}) \cdot \sin(\theta_{new} - \theta_{old})$$

$$y_{new} = y_{center} + (y_{old} - y_{center}) \cdot \cos(\theta_{new} - \theta_{old}) + (x_{old} - x_{center}) \cdot \sin(\theta_{new} - \theta_{old})$$

כאשר `center` משמעותו מרכז המסה של המלבן.

כמו כן, במהלך בניית הממשק נעשה שימוש ב-4 שיטות שונות לשמירת משתנים:

שיטה	פונקציות	יתרונות	שימוש בפועל
base workspace	<code>assignin;</code> <code>evalin</code>	נשמר במעבר בין חלונות ממשק	שליחת מידע בין מסכים של הממשק
GUI handles	<code>guidata</code>	עובר בין פונק' ממשק בלי לעבור לסביבת עבודה אחרת (בניגוד ל-base)	שליחת מידע בין הפונק' של מסך יחיד
global variables	<code>global</code>	ראשון להתעדכן בבאפר של משימות	עצירת כיול במצב לילה
.mat file	<code>save;</code> <code>load</code>	נשמר לטווח ארוך	שמירת הגדרות משתמש

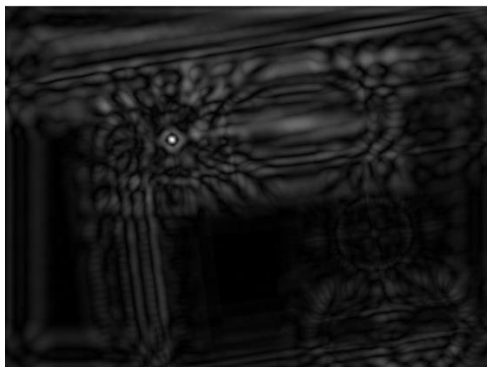
פרק 6: תוצאות

המערכת נבדקה על 33 תמונות לדוגמה שצולמו ע"י הזאולוגים, בהנחה שהן משקפות נאמנה את קלט המערכת האופיני.

זיהוי כרטיס הכיול עובד בהצלחה רבה. מרכז דפוס השמש של כרטיס הכיול זוהה בפיקסל המדויק בו אדם ימקם אותו, בכל התמונות בהן התקיימו התנאים הבאים:

1. הכרטיס הופיע במלואו

2. הכרטיס הופיע במיקוד סביר



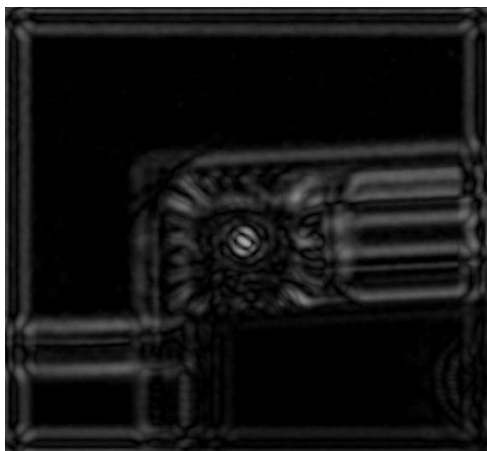
תמונת קורלציה עם גרעיני שמש



חיתוך של תמונת מקור טיפוסית

איור 23 – דוגמה לקורלציה בסביבת דפוס שמש

התמונות בהן לא מתקיימים התנאים ממילא לא מייצגות קלט תקין, שכן הן בניגוד להנחות תחתן פועלת המערכת, לכן שיעור ההצלחה עומד על 100%.



שיא קורלציה נמוך ומקומי

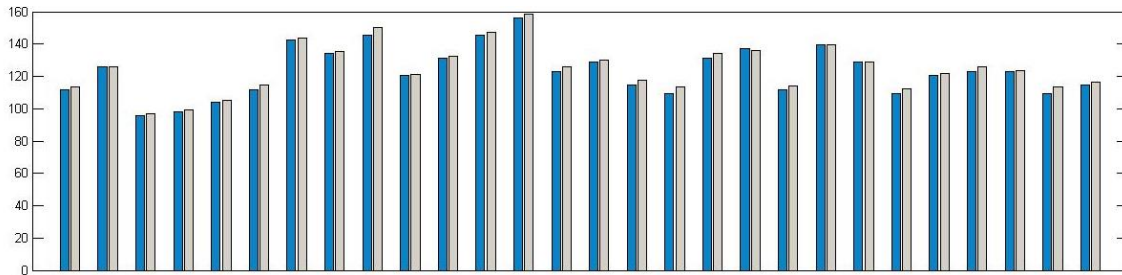


כרטיס מטושטש

איור 24 – דוגמה לקורלציה בסביבת דפוס שמש מטושטש

עם זאת, תהליך מציאת השמש לוקח זמן רב: כ-9 דקות במעבד כפול ליבה בעל קצב שעון של 2.1GHz.

זיהוי זווית כרטיס הכיול עומד בקנה אחד עם התאוריה, ומבטיח סטיה מרבית של 1.2° . זיהוי גודל הכרטיס נבדק על כל התמונות שעברו את שלב זיהוי השמש, והתקבל בדיוק של 98.5% בממוצע, עם סטיה מרבית של 4%.



איור 25 – אורכי כרטיס כיול במציאות (אפור בהיר) ובמערכת (כחול), בפיסקלים

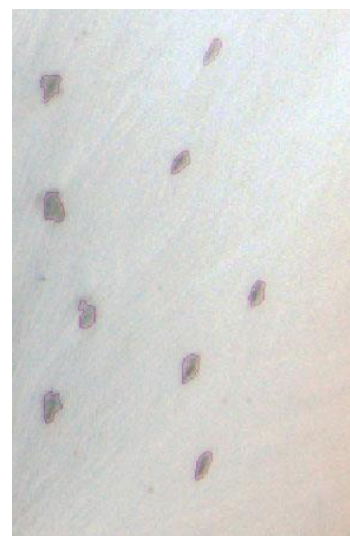
באשר לזמן החישוב, יש לציין שאלגוריתם מציאת זווית וגודל הכרטיס נמשך שניה או שתיים במערכת שצוינה קודם.

הנקודות על גחון התנשמת מזוהות בשיעור של 88% תחת התנאים הבאים:

1. האזור המעובד לא חורג מגבולות התנשמת
 2. האזור המעובד לא כולל אזורים של כתמים כהים שאינם נקודות
- כאשר תנאים אלה לא מתקיימים, יש אובייקטים נוספים ש"שורדים" את פעולת bottom hat שתוארה בחלק 3.3, נשלחים לאלגוריתם k-means, ומשבשים אותו. דוגמה למקרה שכזה מוצגת להלן.



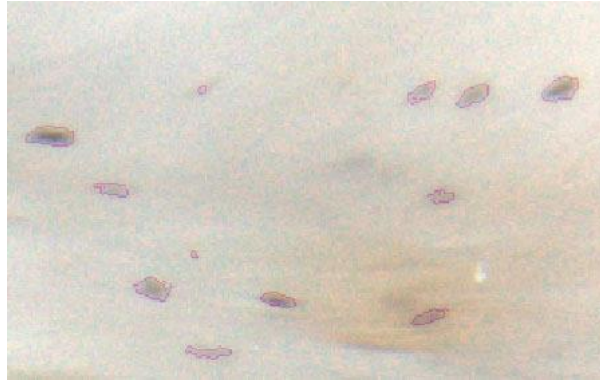
אזור כהה ורועש משבש את הזיהוי



נקודות מזוהות כראוי

איור 26 – השוואה בין זיהוי נקודות בהיעדר כתם מחוספס, ובנוכחותו.

כתם כהה שמשבש את זיהוי הנקודות איננו חייב להיות ככה יחסית לאזור העיבוד. כל שצריך הוא חיספוס של הכתם, ודמיון בין צבעו לבין צבע נקודה. כאשר מדובר בכתם חלק, מנגנון הזיהוי לא משובש. להלן איור לדוגמה.



איור 27 – זיהוי נקודות מוצלח, בנוכחות כתם בחלק התחתון

באזורים כהים של הפלומה, מנגנון זיהוי הנקודות נכשל בגלל הדמיון הרב בין נקודה לפלומה. אלגוריתם k-means לא מצליח לסווג כיאות את הפיקסלים הכמעט-הומוגניים באזור העיבוד. להלן דוגמה:



איור 28 – זיהוי נקודות כושל באזור כהה מאוד

למרות הדרישה לבחירה מדויקת של אזורים, עיבוד תמונה שלמה של תנשמת דורש בממוצע רק 7 סימונים מהמשתמש.

מנגנון הוספת נקודה נבחרת מזהה בהצלחה כ-95% מהנקודות, ויש לו יתרון מובהק עבור נקודות בהירות. באיור להלן ניתן לראות דוגמה לכך.



זיהוי ע"י הוספת נקודות נבחרות

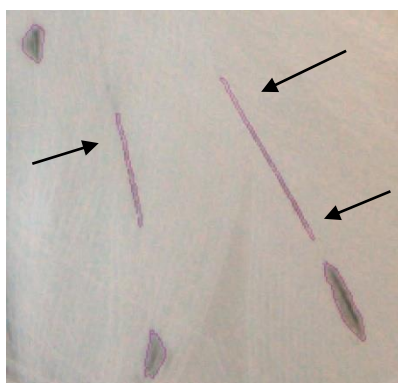


זיהוי נקודות רגיל

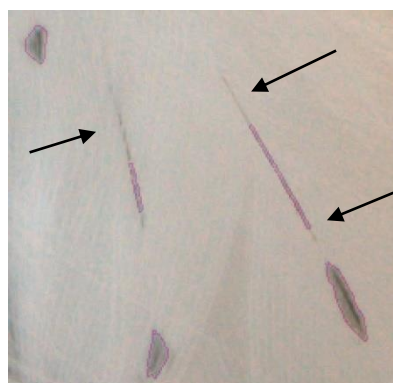
איור 29 – אזור בו הוספת נקודה נבחרת מניבה תוצאות טובות יותר מאשר זיהוי נקודות

הנקודות בתמונה הימנית הושמטו בשלב האחרון של זיהוי הנקודות, שכן הן בהירות בצורה חריגה.

כמו כן, ניתן להשתמש במנגנון זה על מנת לתקן את זיהוי הנקודות ההמוני, ע"י הוספת חלקים שהושמטו מהנקודות. דוגמה באיור להלן.



זיהוי לפי בחירה ע"ג זיהוי רגיל



זיהוי נקודות רגיל

איור 30 – דוגמה לשיפור זיהוי נקודות בעזרת הוספת נקודה נבחרת

בדוגמה לעיל ניתן לראות שהאזורים המסומנים בחיצים הוחמצו ע"י מנגנון זיהוי הנקודות, אך הצליחו להתווסף באופן סלקטיבי ע"י מנגנון הוספת נקודה נבחרת.

פרק 7: מסקנות

במהלך הצילום, יש להקפיד:

- לדאוג שהתמונה ממוקדת
- להשתדל במידת האפשר להביא את כרטיס הצילום לאותו מרחק מהמצלמה כמו הנקודות
- להשתדל ליצור תאורה אחידה של התנשמת

במהלך סימון אזור העיבוד, מומלץ למשתמש:

- להימנע מלכלול אזורים שחורגים מגבולות התנשמת
- להימנע מלכלול כתמים מחוספסים
- להימנע מלכלול אזורים מוצלים במיוחד

הסתייגויות אודות טיב המידע המופק:

- גוון פלומה ממוצע תואם את המציאות רק כאשר התאורה אחידה. בפועל ישנם צללים רבים ומפלי בהירות, הנובעים ממבנה הגלילי של התנשמת.
- בשיטת עיבוד הנקודות הנוכחית, המשתמש מפלג את התנשמת לאזורים. לכן, המרחק המינימלי הממוצע בין נקודות הוא בעל תוקף מקומי בלבד, ואיננו מאפיין של התנשמת השלמה.

במהלך העבודה נוסו דרכים רבות לזיהוי הנקודות על גחון התנשמת. להלן השיטות שלא עבדו, והמסקנות שעולות מכשלונן:

- זיהוי נקודות ע"י גילוי שפות, סגירתן ומילואן – לא עבד כי קיימות הרבה נקודות שיש להן שוליים רכים, שדוהים בהדרגה לצבע הפלומה. כך נוצר מצב שעוצמת הגרדיאנט בשולי הנקודה דומה לעוצמת הגרדיאנט באזורים בפלומה שיש בהם שינויים קלים כמו הצללה.
- זיהוי נקודות ע"י תנאי סף לבהירות – לא עבד כי לעיתים קרובות לנקודות יש רמת בהירות שדומה לזו של פלומה מרוחקת מהן בתמונה. נקודות שכאלה בולטות באופן מקומי בלבד, ואילו תנאי סף פועל בצורה גלובלית על האזור המעובד.

- שיפור מנגנון זיהוי הנקודות בדרכים הבאות:
 - הגברת ניגודיות הקלט – לא תרם דבר
 - תנאי סף בהירות על תוצאת bottom hat – גרם לאלגוריתם לזהות את הנקודות כקטנות מדי.
 - לאחר יצירת המסכה, סילוק אובייקטים שאין בהם גבולות חזקים – פגע בזיהוי נקודות עדינות, ותרם במקרים מועטים.
 - הוספת מיקום הפיקסל כפרמטר ל-k-means מתוך המחשבה שפיקסלים של נקודות קרובים זה לזה גיאוגרפית. ניסיון זה שיבש לחלוטין את התוצאות. הסיבה, כנראה, היא שנקודות הן תמיד יותר קרובות לפלומה מאשר אחת לשניה, לכן מיקום הוא גורם שמערבב את הקבוצות.
 - הפעלת אלגוריתם k-means במרחב בן 27 מימדים, כאשר כל איבר הוא ערכי ה-RGB של פיקסל בתמונה ושל 8 שכניו. הדבר נעשה מתוך המחשבה שנקודות מתאפיינות בצבירי פיקסלים מאותו סוג, ולא פיקסל יחיד. בפועל, הדבר גרם לתוצאת האלגוריתם לעבור LPF ותו לא.
 - בנוסף, נעשה ניסיון לתת לפיקסל המרכזי משקל רב יותר מלשכניו, והדבר גרם לתופעת LPF חלשה יותר מבלי לשפר את התוצאות.
 - המרת התמונה למרחב HSV והתייחסות לרוויה השונה של נקודות לעומת כתמים. ערך הרוויה של כל פיקסל הצטרף ל-RGB שלו באלגוריתם k-means. התוצאות לא השתפרו.
- דפוס השמש שעל כרטיס הכיול הוא בלתי ניתן לזיהוי כאשר הוא לא מצולם במיקוד טוב. מצפיה במקרים של מיקוד ירוד, התגלה כי לא רק שישנו טשטוש, אלא שנוצר דפוס חדש בתדר גבוה במרכז השמש. סביר להניח שניתן להסיק מכאן שמיקוד ירוד גורם לתופעת התחזות (aliasing) בדפוסים מעגליים.

פרק 8: סיכום

המערכת מצליחה לענות על הדרישות שנמנו בפרק המבוא:

- **עמידות** – המערכת מתמודדת היטב עם תאורת התמונות המשתנה ומרחק הצילום המשתנה, ומצליחה לכייל כראוי כל תמונה שצולמה בצורה סבירה. כמו כן, כתמים חלקים על גחון התנשמת לא מהווים מכשול למנגנון זיהוי הנקודות (להראות זאת בתוצאות). עם זאת, כתמים מחוספסים מקשים על המערכת ומאלצים התערבות מדויקת של המשתמש.
- **דיוק** – איכות הנתונים גבוהה, ואולי אף טובה יותר מאלה שאדם היה יכול לחלץ, שכן פיקסלים הם לרוב קטנים יותר משנתות של סרגל. עם זאת, אין מן הנמנע להטיל ספק בתועלתו של גוון פלומה ממוצע.
- **נגישות ממשק משתמש** – הממשק אינטואיטיבי, מאיר פנים, משתמש בסמלים אוניברסליים ובמחוות מוכרות. הודעות שגיאה ואזהרה מיידעות את המשתמש בכל מקום שהן נדרשות, ומדריכות אותו לפתרון. חלון העזרה מפורט ושימושי.
- **זמן עיבוד** – תהליך הכיול לוקח זמן בלתי מבוטל על מחשב דו-ליבתי ממוצע, אך ניתן לעשותו בזמן נוח כרצון המשתמש. כאשר הכיול נעשה מראש, המערכת זריזה ביותר.

המערכת תשמש את המחלקה לזאולוגיה באוניברסיטת תל אביב, תחסוך זמן ומאמץ לחוקרים ותקדם את המחקר לטובת אוכלוסיית התנשמות ובני האדם.

8.1 פיתוח עתידי

להלן מספר הצעות לשיפור המערכת ולמחקר עתידי:

1. המערכת שנבנתה בפרויקט הנוכחי איננה אוטומטית לחלוטין. כלומר, יש צורך באדם שיפעיל אותה צעד-אחר-צעד. הסיבה העיקרית לכך היא שהמערכת לא מזהה את התנשמת, ואזורי העיבוד מוזנים לה ע"י המשתמש. מצב זה לא נותן מענה שלם לחוקרים, שרצונם הליבתי הוא

מיכון וחיסכון בזמן.

לכן הצעה לפיתוח עתידי תהיה הוספת אלגוריתם לזיהוי התנשמת בתמונה. אלגוריתם שכזה יהיה פשוט אם, למשל, ימקמו בזמן הצילום רקע אחיד מאחורי התנשמת.

כך, מרגע שגבולות התנשמת יזוהו, יהיה אפשר להפעיל מנגנון זיהוי נקודות משופר שיודע להתמודד עם כתמים מחוספסים והתוצאות תופקנה ללא התערבות המשתמש.

2. באזורים בהם זיהוי הנקודות מתקשה במיוחד בעבודתו, כדאי להתאים את הממשק כך שיציע מצב של זיהוי נקודות מקומי בלבד. במצב שכזה, לא יופעל זיהוי נקודות המוני, אלא יתבקש מהמשתמש ללחוץ על הנקודות, ורק אלגוריתם 'הוספת נקודה נבחרת', שיש לו אחוזי הצלחה גבוהים במיוחד, יפעל.

3. אלגוריתם מציאת דפוס השמש של כרטיס הכיול משתמש, כאמור בפרקים קודמים, בחישוב קורלציה. חישוב זה הוא איטי למדי, והוא למעשה צרכן הזמן העיקרי של המערכת כולה. במקום להתייחס לכל התמונה, עדיף להתמקד ולחפש את דפוס השמש רק באזורים בעלי רווית צבעים נמוכה בתמונה, שכן כרטיס הכיול מתאפיין בגווי אפור ולכן רווית צבעיו נמוכה. במקום קורלציה, עדיף, אולי, למצוא באזורים אלה אובייקטים בעזרת אלגוריתם כמו watershed, ולבדוק היכן נמצאו 24 עצמים צמודים בעלי שטח דומה – שם נמצאת השמש.

4. מנגנון זיהוי הנקודות רגיש לנוכחותם של כתמים מחוספסים, אך לא לכתמים חלקים. כדאי לבדוק אם בעיה זו נפתרת ע"י העברת הקלט ב-LPF, שיסיר את החספוס מהכתמים.

5. כדאי, אולי, להפוך את אלגוריתם זיהוי הנקודות לאיטרטיבי. לאחר כל פעולה, יבדקו האובייקטים שנוצרו: אלו שיש בהם חורים, ואלו בעלי צורה מעוקלת הם לא נקודות. עם זאת, הם חשודים כנקודות וכנראה מכילים אותן, לכן יתבצע עבורם זיהוי נקודות פנימי.

6. אפיון קצה נוצות התנשמת – פן נוסף במחקר שלשמו נבנתה המערכת.

ביבליוגרפיה

- **Digital Image Processing**
R.C. Gonzalez, R.E. Woods
University of Michigan, 2002.
- **Image Processing and Analyzing**
Michael Elad
Technion, October 2002.
- **Automated Visual Recognition of Individual African Penguins**
T. Burghardt, B. Thomas, P.J. Barham, J. Ali
University of Bristol, Department of Computer Science
September 2004.
- **Mathematical morphology**
Wikipedia, the free encyclopedia
http://en.wikipedia.org/wiki/Mathematical_morphology
- **Exercise 2: Introduction to Binary Images**
Computerized Image and Signal Processing (236327)
Technion, spring 2012.
- **K-means clustering**
MathWorks MATLAB documentation
<http://www.mathworks.com/help/toolbox/stats/kmeans.html>
- **Watershed (image processing)**
Wikipedia, the free encyclopedia
[http://en.wikipedia.org/wiki/Watershed_\(image_processing\)](http://en.wikipedia.org/wiki/Watershed_(image_processing))
- **Barn Owl Studios logo Sample Black**
Angie Mcfarlane
<http://angiemcfarlane.wordpress.com>