

236754

פרויקט במערכות נבונות
פקולטה למדעי המחשב, טכניון

סגמנטציה של תמונות בעזרת ערוצי צבע ועומק

אסף כהן
assa85@gmail.com

הילה ששתיאל
sashila@t2.technion.ac.il

מנחה:
אמיר גבע

תוכן עניינים

3.....	מבוא
3.....	סגמנטציה
3.....	שיטות קיימות
6.....	רקע לפרויקט
6.....	יעדים
7.....	סקירה כללית
7.....	מבנה המערכת
8.....	פירוט המודולים העיקריים
8.....	חישוב תמונת גבולות כללית ותמונת העומק
11.....	איחוד אובייקטים
14.....	מציאת אנשים בתמונה
16.....	מציאת האדם בעל היסטוגרמת הצבע הקרובה ביותר
17.....	תוצאות
17.....	שיטת בדיקה
19.....	תוצאות הטסטים
19.....	ביצועי זמן
20.....	נספחים
20.....	נספח 1 – פונקציית המרה מ- RGB ל- HSL
21.....	נספח 2 – יתרונות השילוב של ערוץ העומק וערוץ הצבע
21.....	יתרון בשימוש של ערוץ הצבע בחישוב גבול
23.....	יתרון בשימוש עומק על מנת לחשב גבול
24.....	נספח 3 – דוגמא לשלבי האלגוריתם
26.....	נספח 4 – הוראות הפעלה ותיעוד הקוד
26.....	מיקום קבצי הקוד
26.....	הוראות הפעלה של התוכנית

מבוא

סגמנטציה

תהליך של חלוקת תמונה דיגיטלית למקטעים (סגמנטים) כאוספים של פיקסלים הנמצאים זה ליד זה. מטרת התהליך היא לפשט או לשנות את הייצוג של התמונה לאוסף של אובייקטים בעלי משמעות שניתן להתייחס אליהם לאחר מכן על ידי אלגוריתמים אחרים לראייה ממוחשבת או לעיבוד תמונה, כגון מציאת גבולות או קווים. התוצאה של התהליך היא אוסף של מקטעים זרים המכסים את התמונה כולה.

ההפרדה נעשית על סמך תכונות של פיקסלים כגון צבע, בהירות, מרקם וכדומה. סגמנטציה תמונה בעלת תפקיד עיקרי בעיבוד תמונה וזיהוי תבניות ושלב בסיסי לקראת ראייה ברמה נמוכה, החיונית לזיהוי אובייקטים, עקיבה, זיהוי פנים ושאר יישומי ראייה ממוחשבת. לדוגמא:

- מערכות רפואיות
 - מציאת גידולים
 - ניתוח מונחה מחשב
 - בניית מבנה אנטומי
- זיהוי אובייקטים בצילומי לוויין
- זיהוי טביעת אצבע

שיטות קיימות

ישנן מספר שיטות המשמשות לסגמנטציה. נציג את העיקריות שבהן.

Thresholding

השיטה הפשוטה ביותר. מבוססת על בחירת ערך סף, לפיו תמונת ה-gray-scale מומרת לתמונה בינארית.

Clustering methods

K-means אלגוריתם

התמונה מחולקת ל-K קבוצות באופן איטרטיבי. כל פיקסל בתמונה משויך לקבוצה באופן כזה שהמרחק למרכז הקבוצה יהיה הקטן ביותר. מרחק מחושב על סמך צבע, עוצמה, מיקום (או שקלול של כל המדדים האלו). התכנסות האלגוריתם מובטחת אולם ייתכן ולא יתקבל פיתרון אופטימאלי (תלוי בבחירת K הקבוצות).

Mean-Shift אלגוריתם

אלגוריתם זה, בהינתן קבוצת נקודות במרחב הדו מימדי, מחזיר את האזור עם הצפיפות הכי גבוהה של נקודות. הרעיון הכללי של האלגוריתם:

- מתחילים מנקודה שרירותית במרחב, ומסתכלים על האזור שמסביב לנקודה.
- בכל פעם מוצאים את מרכז מסת הנקודות באזור עליו אנו מסתכלים, ומזיזים את מרכז הסביבה לשם.

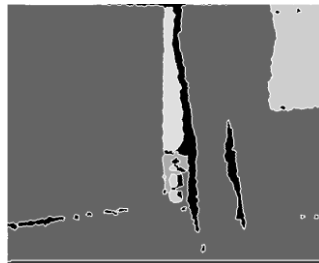
- האלגוריתם ייעצר כאשר מרכז המסה בסביבה שווה לנקודה עליה אנו מסתכלים כעת.

דוגמא לשימוש בשיטה – Edison code (Edge Detection and Image Segmentation System)

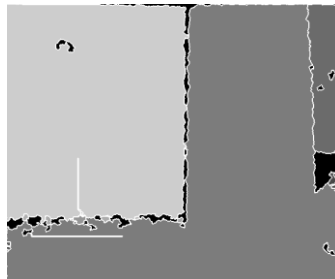
- הפעלת EDISON כאשר שימוש בתמונת צבע בלבד נכשל - לא מצליח להפריד את השרוול (זמן ריצה 5 שניות).



- שימוש בתמונת עומק בלבד נכשל - לא מצליח להפריד את המכנסיים מהרקע (זמן ריצה 3 וחצי שניות).



- הפעלת EDISON כאשר שימוש בתמונת עומק בלבד נכשל - לא מצליח להפריד את האדם מהקיר (זמן ריצה 4 שניות).



- שימוש בתמונת צבע בלבד יוצר הפרדה טובה (זמן ריצה 5 שניות).



שיטות מבוססות היסטוגרמה

שיטות אלו נחשבות יעילות מכיוון שמצריכות מעבר יחיד על הפיקסלים בתמונה. בשיטה זו מחושבת היסטוגרמה מכל הפיקסלים תוך שימוש בצבע ובעוצמה, וערכי השיא משמשים לחלוקת הפיקסלים לקבוצות.

סגמנטציה מבוססת גבולות

גבול – שינוי לוקלי משמעותי בעוצמה בתמונה. גבולות בדרך כלל מופיעים בין שני אזורים בתמונה.

קיימות כמה תת שיטות מסוג זה :

- מציאת המקסימום הלוקלי בסביבת הפיקסל.
- חישוב גבולות – חישוב המקסימום והמינימום המקומיים של הנגזרת הראשונה.

רקע לפרויקט

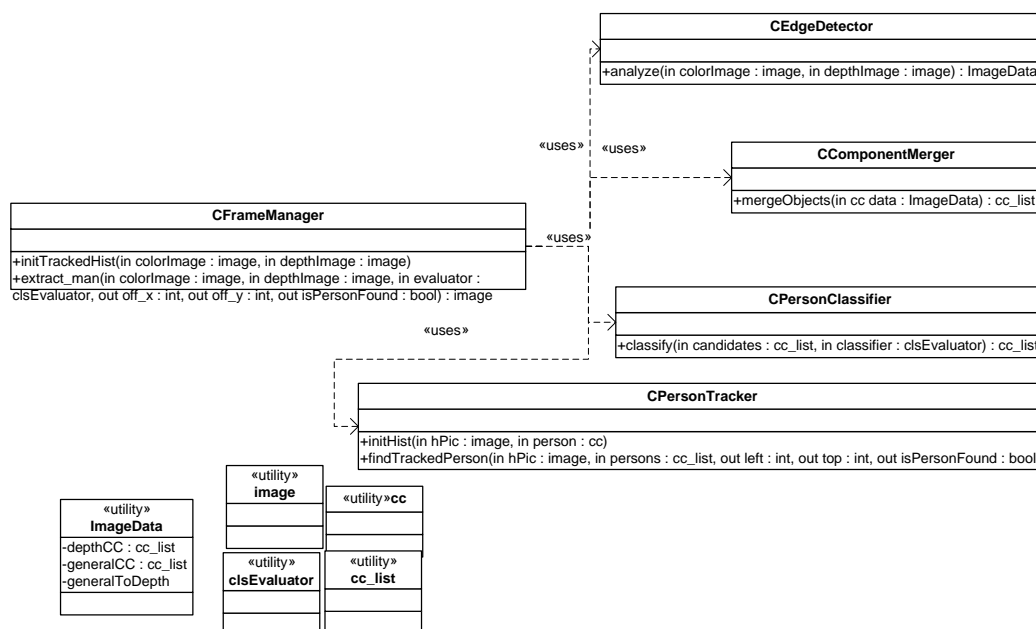
בשנים האחרונות יצאו לשוק מצלמות עם ערוץ עומק (כדוגמת Kinect). מידע זה מסייע במקרים בהם סגמנטציה עפ"י צבע בלבד נכשלת (יפורט בהמשך). הפרויקט ישלב את שני ערוצי המידע במטרה להפיק הפרדת אובייקטים מיטבית.

יעדים

1. סגמנטציה של בני אדם תוך ניצול ערוץ הצבע וערוץ העומק.
 - יכולת הפרדה בין אובייקטים צמודים לפי צבע.
 - יכולת הפרדה בין אובייקטים בעלי צבע זהה לפי עומק.
 2. זיהוי אובייקט מסוים לפי מאפיינים.
 3. זמן ריצה טוב מספיק שיאפשר לרובוט לעקוב אחר בן-אדם בתנועה.
- סדר גודל של 4-15 תמונות בשנייה. סדר גודל זה נראה סביר כיוון שאדם לא יעבור את טווח הראיה של המצלמה בפחות משנייה ובכך נוכל לדגום מספיק תמונות כדי להבטיח עקיבה רציפה של הרובוט.

סקירה כללית

מבנה המערכת



הסברים :

CFrameManager – המחלקה הראשית המייחצנת את שרתי הסגמנטציה של תמונה. מקבלת את תמונת הצבע ותמונת העומק ומחזירה את מיקום האדם לעקיבה בתמונה שהתקבלה.

CEdgeDetector – מחלקה המחשבת את רכיבי הקשירות בתמונה ע"י תמונת הצבע ותמונת העומק ומחזירה מיפוי של כל רכיב קשירות לעומק שלו. פירוט על האלגוריתמים לחישוב זה בסעיף "חישוב תמונת גבולות כללית ותמונת העומק".

CComponentMerger – מחלקה המאחדת רכיבי קשירות שנוצרו ע"י המחלקה CEdgeDetector השייכים לאותו אובייקט. פירוט על האלגוריתמים לחישוב זה בסעיף "איחוד אובייקטים".

CPersonClassifier – מחלקה המקבלת את רכיבי הקשירות הסופיים ומוציאה מהם את רכיבי הקשירות המייצגים את האנשים בתמונה. פירוט על האלגוריתמים לחישוב זה בסעיף "מציאת אנשים בתמונה".

CPersonTracker – מחלקה המאותחלת ע"י רכיב קשירות של הבן אדם לעקיבה ושומרת את היסטוגרמת הצבע שלו. בכל פעם שתקבל מהמחלקה CPersonClassifier את רשימת האנשים בתמונה תחזיר את מיקום הרכיב שמייצג את האדם לעקיבה. אם לא נמצא כזה תחזיר תשובה שלילית. פירוט על האלגוריתמים לחישוב זה בסעיף "מציאת האדם בעל ההיסטוגרמת צבע הקרובה ביותר".

ניתן לראות דוגמא של שלבי האלגוריתם בנספח 3.

פירוט המודולים העיקריים

חישוב תמונת גבולות כללית ותמונת העומק

העיקרון הבסיסי למציאת גבול של אובייקט הוא העובדה שיש מעבר חד של ערכים בין פיקסלים שכנים.

על מנת שערכו של פיקסל ייצג את צבעו השתמשנו במרחב צבע HSL שהומר מהמרחב RGB שבו קיבלנו את התמונה (נוסחת ההמרה בנספח 1). משמעות מרחב ה-HSL:

H – Hue – גוון הצבע של האובייקט. מיוצג כזווית בתחום $[0, 2\pi]$.

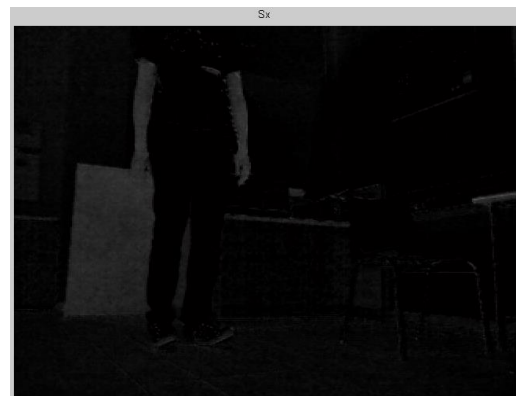
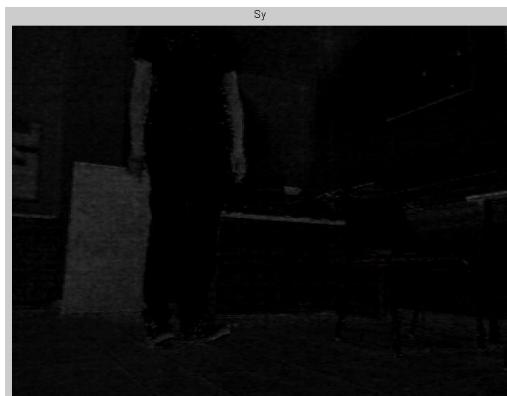
S – Saturation – עד כמה הצבע רווי.

I – Lightness – עד כמה האובייקט מואר.

ניתן לראות כי מהגדרת מרחב זה מדדים אלו משקפים טוב יותר את האובייקטים בתמונה (מאשר RGB).

לאחר ההמרה למרחב זה העברנו את S לקואורדינטות קרטזיות. זאת על מנת שנוכל לבטל מצבים בהם S קטן ו-H משתנה משמעותית. במצבים כאלו כאשר S קטן אין הבדלים בין גוונים של צבע (הצבע אפור) ולכן לא נרצה להתייחס ל-H.

לכן המרחב הסופי שקיבלנו: $S_x = S \cdot \cos(H)$, $S_y = S \cdot \sin(H)$, I.



בנוסף לערוץ הצבע השתמשנו בערוץ עומק.



זאת כיוון שהוא מעניק מידע המשלים את הצבע במקרים בהם שני אובייקטים דומים מבחינת צבע אך רחוקים זה מזה ולכן ערוץ העומק יעניק את הגבול הרצוי. מצד שני ערוץ הצבע מעניק מידע כאשר אדם קרוב מאוד לאובייקט אחר ואין ביניהם הבדלי עומק. דוגמאות עם תמונות בנספח 2.

לאחר שקיבלנו את מרחב הצבע הרצוי עלינו לחשב לפי מרחב זה וערוץ העומק את ההבדלים בין ערכי הפיקסלים. לשם כך חישבנו לכל פיקסל את מרחקו האוקלידי במרחב ה-4 מימדי ($S_x = S \cdot \cos(H)$, $S_y = S \cdot \sin(H)$, עומק) מהפיקסלים שמעליו ומצדדיו, וקבענו את ערכו בתור המרחק המקסימאלי. בחרנו להשתמש בשכנים במרחק 2 כלפי מעלה ומטה ולצדדים כי מולם יהיה מרחק משמעותי ויאפשר לזהות מעבר חד. כלומר ערכו של כל פיקסל הוא:

$$\max\{dist(up), dist(down), dist(right), dist(left)\}$$

כאשר מרחק מחושב על פי הנוסחה:

$$dist(p_1, p_2) = (p_1.l - p_2.l)^2 + (p_1.sx - p_2.sx)^2 + (p_1.sy - p_2.sy)^2 + factor * (p_1.depth - p_2.depth)^2$$

ההכפלה של הגורם של הפרש העומק בפאקטור הוא כדי לתת משקל שווה לעומק לעומת הצבע, כלומר ערכו המספרי של הפרש העומק יהיה דומה לערכים המספריים של סכום הפרשי הצבע.

לאחר שחישבנו את המרחק של כל פיקסל משכניו נרמלנו את כל המרחקים לפי המרחק המקסימאלי בתמונה והפכנו לבינארי.

התוצאה:



יחד עם זאת אנו מחשבים את הגבול של תמונת העומק. זאת על מנת שנוכל להשתמש בקביעת עומקו של רכיב קשירות בשלב איחוד רכיבי הקשירות (יוסבר בהרחבה בסעיף הבא). בתמונת העומק לרוב הרצפה מחוברת לאנשים כלומר אין גבול בין האנשים לרצפה.

דוגמא :

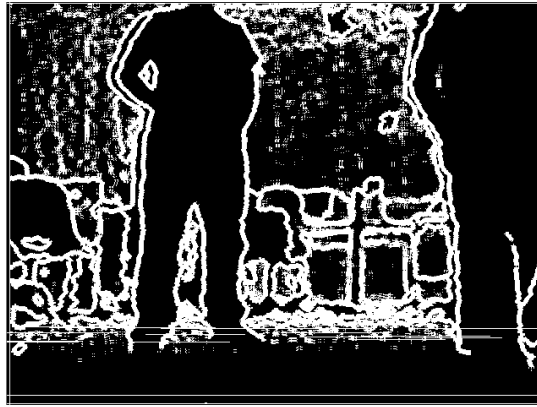


ניתן לראות כי הרצפה היא אזור שחור ולרוב רגלי האנשים הם האזור גבול (פיקסלים לבנים) הראשון. לכן הגדרנו אלגוריתם בסיסי למציאת תחילת החלק התחתון של האנשים:

1. עבור החלק התחתון של תמונת הגבול סיכמנו עבור כל שורה את מספר הפיקסלים שיש בה
2. השורה התחתונה ביותר בעלת כמות פיקסלים לבנים מעל רף מסוים הוגדרה כקו התחתון
3. הוגדר מקו תחתון זה אזור של כמות שורות קבועה כלפי מעלה שבו אמור להיות תחילת רגלי האנשים
4. עבור אזור זה סיכמנו את העמודות ושמרנו את העמודות שסכומם עולה בקבוע מהעמודות שמימנם ומשמאלם, נגדירים כ"שיאים". זאת כיוון שלרוב תחילת הגבול של רגל של אדם היא שני עמודות של קווים לבנים רחוקים זה מזה :



5. הסרנו את השיאים שמרחקם משיאים שכנים הוא מעל קבוע המגדיר רוחב של רגל אדם
6. לבסוף ציירנו קו בין השיאים לקבלת סגירת המרווח של הרגלים :



איחוד אובייקטים

לאחר חלוקת התמונה לרכיבי קשירות, נוצר מצב בו אובייקט בודד מופרד לכמה רכיבים כתוצאה מהבדלי צבע.

חלוקה זו עלולה לגרום לבעיה בשלב זיהוי האנשים בתמונה (ה-classifier לא יקבל את אובייקט האדם בשלמותו) ובשלב חישוב ההיסטורגמה (תתקבל היסטורגמה חלקית בלבד). לכן, יצרנו מנגנון שתפקידו לאחד רכיבי קשירות ששייכים לאותו האובייקט. על מנת לקבוע האם לאחד שני רכיבים התחשבנו במספר תנאים:

- עומק.
- יחס הכלה.
- יחס חיתוך-איחוד.

ראשית דרשנו ששני האובייקטים יהיו בעלי עומק זהה. לצורך כך נעזרנו בחישוב רכיבי הקשירות שהתקבלו כתוצאה משימוש בתמונת העומק בלבד. תמונת הרכיבים הקשירים שמתקבלת משילוב של תמונת הצבע ותמונת העומק היא למעשה עידון של תמונת הרכיבים הקשירים שמתקבלת על סמך תמונת העומק בלבד. לכן, יצרנו מיפוי שמטרתו לשייך אובייקט מסוים לאובייקט המתאים לו בתמונת העומק. שיוך של אובייקט O_{total} לאובייקט העומק O_{depth} התבצע על סמך שני קריטריונים:

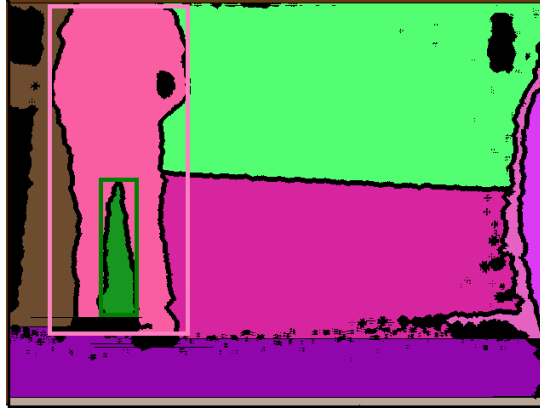
1. היחס בין חיתוך שני המלבנים החוסמים של O_{total} ו- O_{depth} לבין המלבן החוסם של O_{total} .
2. היחס בין חיתוך שני המלבנים החוסמים של O_{total} ו- O_{depth} לבין המלבן החוסם של O_{depth} .

השיוך של אובייקט O_{total} כלשהו לאובייקט O_{depth1} יועדף על פני שיוכו לאובייקט O_{depth2} אחר, במידה והיחס הראשון גדול יותר.

כאשר היחס הראשון זהה (או כמעט זהה), נעדיף את האובייקט שנותן יחס שני גדול יותר.

שני הקריטריונים נחוצים משום שההשוואה מתבצעת על סמך המלבנים החוסמים של האובייקטים. לכן, יתכן מצב בו אובייקט מסוים יהיה מוכל לגמרי בשני רכיבי עומק (וכתוצאה מכך היחס הראשון יהיה 1), אולם נעדיף לשייכו לרכיב העומק הקטן מביניהם שמכיל אותו (משום שהאחרון יניב יחס שני גדול יותר).

לדוגמא (תמונה 124562), נסתכל ברכיבי הקשירות שנוצרו מתמונת העומק (כל רכיב קשירות צבוע בצבע שונה).



כעת, נתבונן ברכיבי הקשירות שנוצרו מתמונות העומק ותמונת הצבע. ניתן לראות את מיפוי האובייקטים לרכיבי עומק (כל אובייקט צבוע בצבע של רכיב העומק אליו מופה). נפעיל את אלגוריתם השיוך על האובייקט הירוק.



המלבן החוסם של אובייקט זה מוכל במלואו במלבן החוסם של רכיב העומק הוורוד ובמלבן החוסם של רכיב העומק הירוק, אך מכיוון שרכיב העומק הירוק קטן יותר נמפה אליו (וכך נמנע בעתיד איחוד של האובייקט הירוק לאדם הוורוד שיתמפה לרכיב העומק הוורוד).

נציין, שהשוואה של הפיקסלים של האובייקטים עצמם ללא שימוש במלבנים חוסמים לא התאפשרה משום שלא עמדה במגבלות ביצועי הזמן.

לכן, שני מועמדים לאיחוד יהיו בעלי עומק זהה אם שויכו לאותו רכיב עומק.

לאחר שבדקנו ששני המועמדים שווים בעומק, דרשנו שיתקיים אחד מהשניים:

- יחס הכלה גדול מסף מסוים.
- יחס חיתוך-איחוד גדול מסף מסוים.

בהינתן שני מועמדים לאיחוד O_{total1} ו- O_{total2} , יחס הכלה חושב כיחס בין המלבן החוסם של החיתוך בין שני האובייקטים לבין המלבן החוסם של האובייקט הקטן יותר. יחס זה מהווה מדד לאחוז ההכלה של האובייקט הקטן באובייקט הגדול.

$$\text{Containment_Ratio} = \frac{O_{total1} \cap O_{total2}}{O_{total2}}, |O_{total2}| < |O_{total1}|$$

יחס חיתוך-איחוד חושב כיחס בין רוחב המלבן החוסם של החיתוך בין שני האובייקטים לבין רוחב

המלבן החוסם של האיחוד בין שני האובייקטים. בנוסף דרשנו ששני האובייקטים יהיו אחד מעל השני. יחס זה מהווה מדד לאחוז השוויון ברוחב האובייקטים.

$$\text{Intersection_Union_Ratio} = \frac{O_{total1}.width \cap O_{total2}.width}{O_{total1}.width \cup O_{total2}.width}$$

נביא דוגמא (תמונה 320437 במקבץ 3).

בתמונה ניתן לראות את האובייקטים לפני תהליך האיחוד (כל אובייקט צבוע בצבע שונה).



לאחר תהליך האיחוד קיבלנו את תמונת האובייקטים הבאה:



ניתן לראות שהרגל הורודה אוחדה למכנסיים הירוקות כתוצאה מיחס הכלה. לאחר מכן, החולצה הורודה אוחדה למכנסיים הירוקות כתוצאה מיחס חיתוך-איחוד. מזכיר, כי תחילה בדקנו שלכל האובייקטים עומק זהה. בתום התהליך קיבלנו את אובייקט האדם בשלמותו.

מציאת אנשים בתמונה

בשלב זה, נתונים בידינו האובייקטים שחולצו מהתמונה, מתוכם אנו רוצים למצוא את האובייקטים שמייצגים אנשים.

לשם כך השתמשנו ב-classifier שמומש באמצעות טכניקת SVM (Support Vector Machin). בעזרתה ניתן, באמצעות למידה מונחית, לנתח ולזהות תבניות שימשו לסיווג אובייקטים.

טכניקה זו כוללת שלב אימון שתכליתו יצירת מסווג נכון ככל האפשר בין דוגמאות אימון חיוביות ושליליות.

כדי לייצר את דוגמאות האימון השתמשנו בתוכנית הבדיקה (שתפורט בהמשך). אובייקטים שסווגו כאנשים בהצלחה נלקחו לדוגמאות החיוביות, ויתר האובייקטים נלקחו לדוגמאות השליליות. נעשה שימוש ב-kernel גאומטרי על מנת ליצור הפרדה לא לינארית.

סיווג האובייקטים נעשה באמצעות שני מאפיינים:

- Rays Feature – מוציא כפלט ווקטור באורך 60, כאשר כל תא בווקטור מייצג אורך קרן ממרכז הכובד של האובייקט להיקף הצורה בזווית מסוימת. הקרניים מחושבות בהפרשים של 6 מעלות ומנורמלות לאורך של 1 על מנת שהמאפיין לא יושפע ממרחק האדם מהמצלמה.



- Legs Feature – מחשב ווקטור באורך מספר השורות באובייקט, כאשר כל תא בווקטור מייצג את מספר ה-runs בשורה. run מוגדר כרצף פיקסלים לבנים שתחום ע"י פיקסלים שחורים. לדוגמא, עבור האובייקט הבא:



בשורה הראשונה נקבל run אחד ובשורה השנייה נקבל שני runs. הפלט של ה-feature הוא 3 הערכים הבאים:

1. ערך חציון של ה-runs עבור השליש העליון של התמונה (עבור אובייקט המייצג אדם ערך זה יהיה קרוב ל-1).
2. ערך חציון של ה-runs עבור השליש התחתון של התמונה (עבור אובייקט המייצג אדם ערך זה יהיה קרוב ל-2).
3. ערך מקסימאלי של runs שהתקבל עבור האובייקט כולו (מהווה מדד לכמות הרעש).

הקלט לתוכנית האימון של ה-classifier הוא ווקטור באורך מספר תמונות האימון, כאשר כל תא בו מכיל את אוסף ה-features שחושבו עבור האובייקט (עבור כל אובייקט 60 ערכים מ-Rays ו-3 ערכים מ-Legs). הפלט הוא שני ווקטורים באורך מספר תמונות האימון (n):

- $\{X_i, Y_i\}_n$ - כאשר X_i מכיל את ערכי ה-features עבור האובייקט ה- i ו- Y_i מכיל את ציון ההתאמה של האובייקט ה- i . ציון של 1 יציין התאמה וציון של -1 יציין חוסר התאמה.
- $\{\alpha_i\}_n$ - כאשר α_i מהווה מדד לחשיבותה של הנקודה ב-training set על מנת להגדיר גבול (גבוה ככל שגדלה החשיבות).

ה-classifier מקבל אובייקטים החשודים כאנשים (לפני קריטריונים של גודל, רוחב וגובה) ומחשב עבור כל מועמד ציון התאמה.

חישוב הציון נעשה על סמך תוצאות האימון באופן הבא:

X_{new} - ווקטור ה-features עבור האובייקט החדש הנבדק.

b - קבוע שמטרתו להזיז את הגבול לנקודה 0.

k - פונקציית kernel גאוסית: $k = e^{-\frac{\|X_i - X\|^2}{2\sigma^2}}$, כאשר σ קבוע כלשהו.

$$Grade = b + \sum_i \alpha_i Y_i k(X_i, X)$$

ציון חיובי יינתן למועמד שמזוהה כבן-אדם. אחרת יינתן לאובייקט ציון שלילי.

המועמדים בעלי הציונים החיוביים עוברים לשלב הבא באלגוריתם.

מציאת האדם בעל היסטוגרמת הצבע הקרובה ביותר

בשלב זה, נתונים בידנו מספר מועמדים לעקיבה מתוכם אנו רוצים לבחור את המועמד המתאים ביותר.

לשם כך, ביצענו קוונטיזציה על ערוץ ה-H (שנגזר מתמונת ה-RGB) עבור כל אחד מהאובייקטים, ובחרנו את המועמד בעל ההיסטוגרמה הקרובה ביותר להיסטוגרמה הרצויה.

הקוונטיזציה בוצעה באופן יוניפורמי, כאשר מרחב ה-H חולק ל-7 נתחים שווים. כל פיקסל השייך לאובייקט הנבחן משויך לאחד הנתחים על-פי ערכו. היסטוגרמת הצבע מייצגת את אחוז הפיקסלים בנתח מסוים מתוך כלל הפיקסלים של האובייקט. השתמשנו במדד זה מכיוון שמספר אבסולוטי לא יספק פיתרון למצב בו האדם מתקרב או מתרחק מהמצלמה.

כדי לבחור את המועמד המתאים ביותר, ביצענו שלב של אתחול ההיסטוגרמה הרצויה. שלב זה מתבצע באתחול אלגוריתם העקיבה של הרובוט (כפתור מיוחד). על סמך התמונה שמתקבלת אנו מחלצים את האדם בתמונה ומבצעים עבורו את תהליך יצירת ההיסטוגרמה שתואר. את ההיסטוגרמה שהתקבלה אנו שומרים, ולפיה יתבצעו כל ההשוואות העתידיות.

על מנת למצוא את האדם בעל ההיסטוגרמה הקרובה ביותר להיסטוגרמה הרצויה אנו מחשבים מרחק אוקלידי, כאשר כל אחד מ-7 הנתחים מהווה מימד, ובחרים את ההיסטוגרמה בעלת המרחק הקצר ביותר.

במידה וכל המרחקים שחושבו עבור המועמדים גדולים מסף מסוים, האלגוריתם יחזיר תשובה שלילית שמשמעותה שבתמונה הנוכחית אין את האדם שאנו עוקבים אחריו.

תוצאות

שיטת בדיקה

התוכנית נבדקה בשני שלבים:

- בשלב הראשון נבדקה יכולת בניית תמונות הגבולות וחילוץ האובייקטים מתוכה.
- בשלב השני נבדקה יכולת מציאת האדם המתאים ביותר לעקיבה.

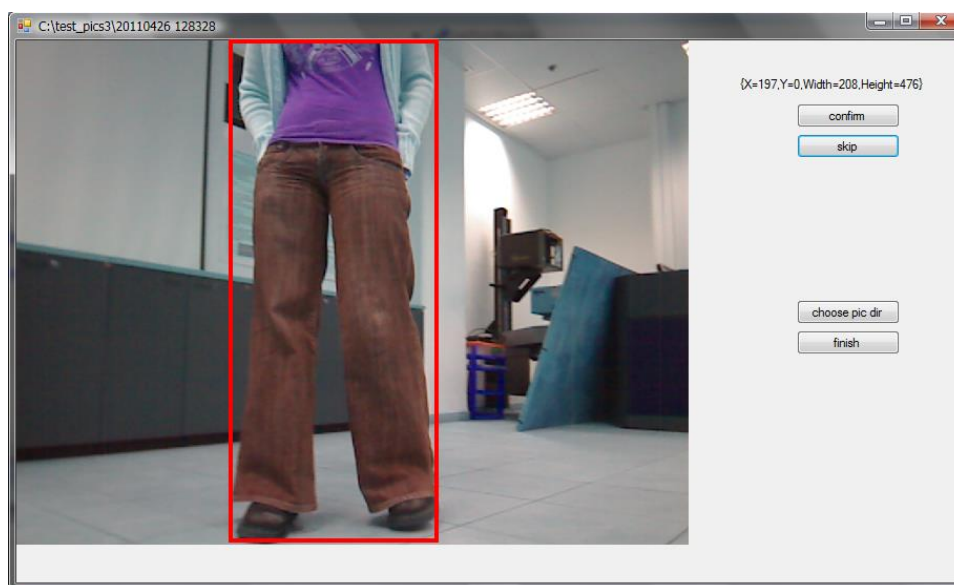
תיאור השלב הראשון:

תחילה יצרנו מאגר תמונות מבחן שכולל תמונות צבע ותמונות עומק. התמונות הוקלטו ע"י שימוש ברובוט כאשר גובה המצלמה נבחר כך שיתקבלו התוצאות הטובות ביותר.

בכדי לבחון את יכולת חילוץ האובייקטים הכנו קובץ תוצאות צפויים שיכיל, עבור כל תמונה שנבדקה, את המיקום הצפוי של האדם בתמונה.

המיקום הוגדר ע"י 4 פרמטרים: קואורדינטת X, קואורדינטת Y, רוחב המלבן החוסם של האובייקט, אורך המלבן החוסם של האובייקט.

הקובץ יוצר באמצעות כלי סימון שהכנו. כלי זה מבצע מעבר על תמונות המבחן. עבור כל תמונה מאפשר הכלי למשתמש לסמן מלבן סביב אובייקט מסוים, ולאחר סימון האובייקט כותב לקובץ את הפרמטרים הדרושים.

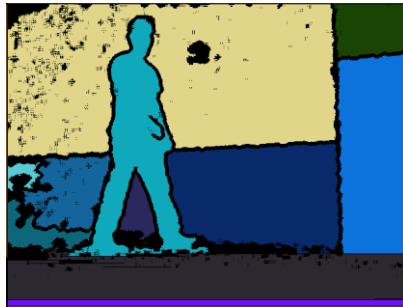


הרצנו את האלגוריתם למציאת רכיבי הקשירות.

כל רכיב קשירות הושווה לאובייקט המסומן בקובץ התוצאות ונמדד עבורו מרחק. מרחק בין אובייקטים מוגדר כיחס בין המלבן החוסם של חיתוך האובייקטים לבין המלבן החוסם של איחוד האובייקטים. יחס זה מהווה מדד להתאמה בין האובייקטים (מבחינת מיקום בתמונה וגודל). לאחר מציאת האובייקט הקרוב ביותר, בדקנו האם מרחקו לאובייקט המצופה קטן מסף מסוים. במידה וכן, התמונה סומנה כמוצלחת. אחרת, התמונה סומנה ככישלון.

בכדי לאפשר debug, שמרנו את התוצאות הבאות עבור כל תמונה:

- רכיבי קשירות שנוצרו מתמונת העומק:



- רכיבי קשירות סופיים שהתקבלו כתוצאה מתמונת העומק ותמונת הצבע לאחר הפעלת אלגוריתם האיחוד:



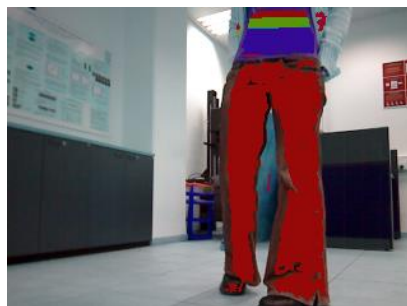
- מיפוי האובייקטים לרכיבי עומק:



בתום מעבר על כל תמונות המבחן הצגנו את אחוזי ההצלחה.

תיאור השלב השני:

בשלב זה בדקנו את יכולת העקיבה אחר אדם מסוים. הגדרנו 2 תמונות אתחול (צבע ועומק) בהן מופיע האובייקט אחריו אנו רוצים לעקוב, לפיהן יתבצע אתחול ההיסטוגרמה. לאחר מכן, הרצנו את האלגוריתם בשלמותו, וסימנו את האובייקט שנבחר לעקיבה. סימון האובייקט התבצע תוך שימוש בהיסטוגרמת הצבע שלו.



תוצאות הטסטים

תוצאות השלב הראשון:

כאשר קובעים את המדד למרחק בין אובייקטים כ- 0.6 מקבלים 70% אחוזי הצלחה. הסיבות לבחירת קבוע של 0.6 ולא 1 הן:

- קושי לשייך נעליים וידיים לאובייקט.
- כאשר אדם קרוב למצלמה – רואים שתי רגליים שאין ביניהן שטח משותף (לכן לא ניתן לאחד ביניהן).



- חישוב תמונות הגבולות מתבצע על תמונה מוקטנת פי 2 תוך שימוש ב-2 שכנים, מה שיוצר גבול רחב (הסיבה להקטנת מימדי התמונה היא לצורכי שיפור ביצועי זמן ריצה).

מעבר ידני על קובץ התוצאות שסומנו ככישלון מראה כי רובן מכילות אובייקטים של בני אדם שה classifier יזהה אותן בהצלחה.

תוצאות השלב השני:

ניתן לראות כי ברוב התמונות האובייקט שסומן הוא האובייקט עליו ביצענו את העקיבה, גם במקרה של מספר אנשים בתמונה.

בדוגמאות ניתן לראות כי האובייקט שסומן הוא האובייקט לפיו התבצע אתחול ההיסטוגרמה.

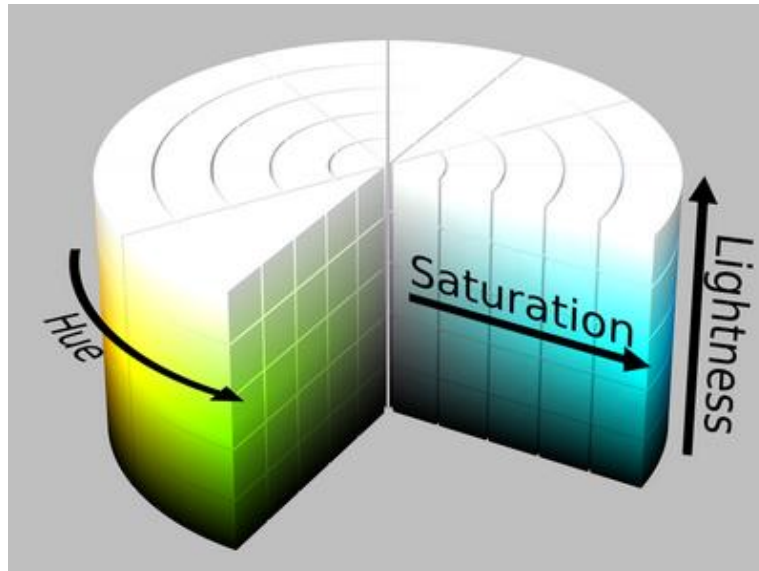


ביצועי זמן

- על המחשב של הרובוט (ATOM): כ-4 תמונות בשנייה.
- על מחשב נייד (2GHz CPU, 2.5GB RAM): כ-20 תמונות בשנייה.

נספח 1 – פונקציית המרה מ- RGB ל- HSL

מרחב HSL בצורה גרפית :



הנוסחה להמרה מ- RGB ל- HSL :

$$H = 2\pi * \begin{cases} 0 & \max_{RGB} - \min_{RGB} = 0 \\ 1 + \frac{(G - B)}{6 * (\max_{RGB} - \min_{RGB})} & \max_{RGB} = R \\ \frac{1}{3} + \frac{(B - R)}{6 * (\max_{RGB} - \min_{RGB})} & \max_{RGB} = G \\ \frac{2}{3} + \frac{(R - G)}{6 * (\max_{RGB} - \min_{RGB})} & \max_{RGB} = B \end{cases}$$

$$S = \begin{cases} 0 & \max_{RGB} = 0 \\ \max_{RGB} - \min_{RGB} & else \end{cases}$$

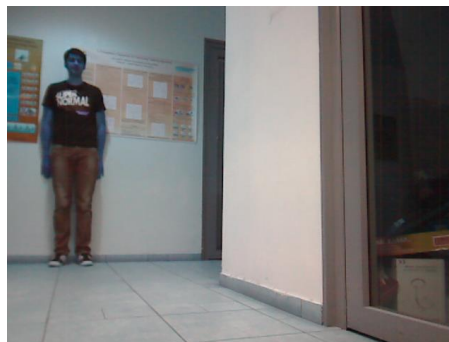
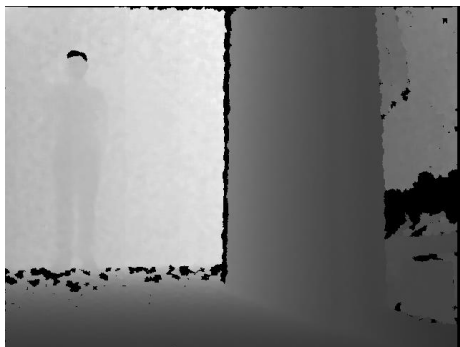
$$L = \frac{(R + G + B)}{3}$$

נספח 2 – יתרונות השילוב של ערוץ העומק וערוץ הצבע

יתרון בשימוש של ערוץ הצבע בחישוב גבול

דוגמא 1 – אדם צמוד לקיר

תמונות הקלט (צבע ועומק):



תמונות הגבולות שמתקבלת ללא שימוש בצבע:



כיוון שהאדם צמוד לקיר אין מספיק הבדלי עומק בינו לבין הקיר ולכן לא נוצרה תמונת גבולות תקינה.

תמונת הגבולות שמתקבלת עם שימוש בצבע :



דוגמא 2 – שני אנשים צמודים

תמונות הקלט (צבע ועומק):

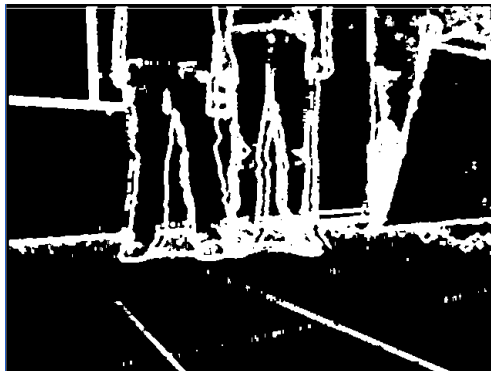


תמונות הגבולות שמתקבלת ללא שימוש בצבע:



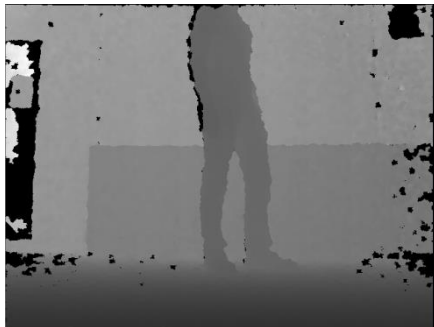
ניתן לראות כי האנשים ללא גבול אנכי ביניהם.

תמונות הגבולות שמתקבלת עם שימוש בצבע:

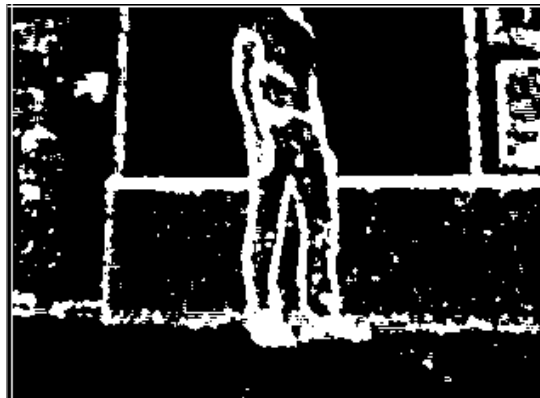


יתרון בשימוש עומק על מנת לחשב גבול

תמונות הקלט (צבע ועומק):



תמונות הגבולות שמתקבלת ללא שימוש בעומק:

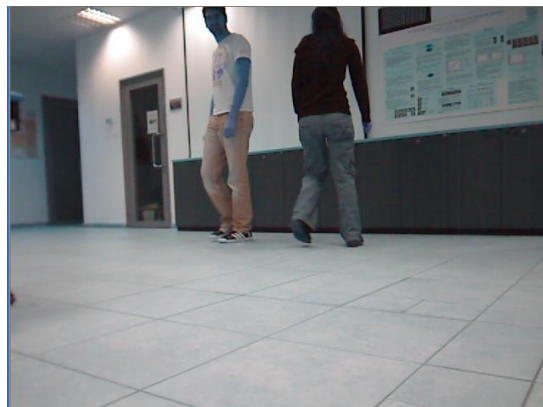


ניתן לראות כי באזור הכתפיים אין הפרדה (אין הבדלי צבע משמעותיים).

תמונות הגבולות שמתקבלת עם שימוש בעומק:



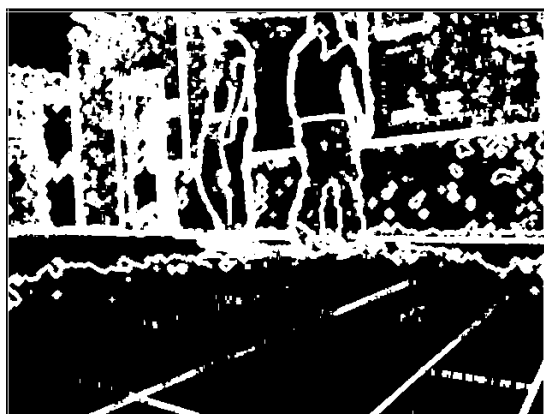
תמונות קלט:



שלב 1: אתחול המערכת בהיסטוגרמה של האדם מימין

שלב 2:

חישוב תמונת גבולות:



חישוב תמונת רכיבי קשירות:



מיפוי רכיבי קשירות כללים לצבע:



שלב 3: איחוד רכיבי קשירות



שלב 4: מציאת האנשים בתמונה




שלב 5: החזרת הקורדינטות של האדם מימין

נספח 4 – הוראות הפעלה ותיעוד הקוד

מיקום קבצי הקוד

1. תחת תיקיית C:\Shared\src
a. extract_man_lib – ספריה המכילה את האלגוריתם לסגמנטציה ועקיבה
i. CComponentMerger.h/cpp
ii. CEdgeDetector.h/cpp
iii. CFrameManager.h/cpp
iv. CPersonClassifier.h/cpp
v. CPersonTracker.h/cpp
vi. InputConstants.h
▪ כל קובץ אחראי על החלק באלגוריתם בהתאם לשמו (כפי שתואר בסעיף של מבנה המערכת)
b. Chase – ספריה המכילה את הקוד של הרובוט המשתמשת בספריית extract_man_lib לקבלת שירותי עקיבה

הוראות הפעלה של התוכנית

1. בשימוש בשלט :
 - a. הפעלת האפליקציה – לחץ 
 - b. אתחול הבן אדם לעקיבה (האדם לעקיבה יעמוד לבדו מול הרובוט) – לחץ 4
 - c. סגירת אפליקצית הרובוט – לחץ clear
 - d. השהיית הרובוט (קופא במקום) – לחץ 2
 - e. לעבור למצב בו הרובוט שומר את התמונות צבע ועומק שהוא קולט – לחץ 3
2. התחברות מרחוק
 - a. התחברות דרך VNCViewer –
 - i. Ip : 192.168.1.102
 - ii. שם משתמש : atom
 - iii. סיסמא : atomvnc