

הטכניון - מכון טכנולוגי לישראל  
הפקולטה להנדסת חשמל ע"ש אנדרו וארנה ויטרבי  
המעבדה לראייה ומדעי התמונה

דו"ח סיכום פרויקט: א'

# מיפוי מסלול בסביבה מוכרת

## Indoor Navigation with Camera

מבצעים:

Izar Hasson  
Raz Shemesh

איזאר חסון  
רז שמש

Chen Katz

מנחה: חן כץ

סמסטר רישום: חורף תשפ"ג

תאריך הגשה: ינואר, 2025

# תוכן עניינים

1.	מבוא	1
2.	סקירת ספרות	2
3.	הפתרון	3
3.1.	תכנון ראשוני	3
3.2.	תיאור הפתרון	3
3.3.	תהליך הפיתרון	4
3.4.	תוצאות	7
4.	סיכום	10
11.	רשימת מקורות	11

## רשימת איורים

- איור 1: דוגמה לחופשיות קנה מידה ..... 1
- איור 2: שלבי המסלול ..... 3
- איור 3 : כל הווקטורים בבעיה ..... 4
- איור 4: משמאל לימין, מסלול ראשון, עשיר פיטצ'רים ..... 7
- איור 5 : פלט ORB-SLAM3, מסלול יחסי בלבד ובלי גודל אמיתי ..... 7
- איור 6 : פלט תהליך זיהוי ומציאת מיקום מצלמה לפי הסמנים ..... 7
- איור 7 : פלט סופי, המסלול בגודל אמיתי עם התייחסות למפה (איקסים הם הסמנים) ..... 8
- איור 8: משמאל לימין, מסלול שני, דל פיטצ'רים ..... 8
- איור 9: פלט תהליך זיהוי ומציאת מיקום מצלמה לפי הסמנים, קיים רק בתחילת וסוף מסלול ..... 9
- איור 10: פלט ORB-SLAM3, מסלול יחסי בלבד ובלי גודל אמיתי ..... 9
- איור 11: פלט סופי, המסלול בגודל אמיתי עם התייחסות למפה (איקסים הם הסמנים) ..... 9

# תקציר

מטרת הפרויקט היא לפתח מערכת חזקה לשחזור מסלול תנועה תלת-ממדי בהתבסס על נתוני וידאו שצולמו במהלך ניווט ותזוזה בסביבה. הפתרון המוצע משלב גישה היברידית שמשתמשת במידע חזותי שמופק מסמנים חזותיים מסוג ArUco כאשר הם נראים במצלמה, ובמערכת ORB SLAM להערכת מיקום כאשר הסמנים אינם נראים במצלמה.

סמני ArUco מספקים הערכת מיקום אמינה, כולל אוריינטציה ומיקום, בכל פעם שניתן לזהותם. כאשר לא ניתן לזהות את הסמנים, מערכת ה-ORB SLAM ממשיכה באופן חלק לבצע מעקב ומיפוי של הסביבה. על ידי שילוב נתוני המיקום שמופקים מהסמנים כנתוני אמת עבור מערכת ה-ORB SLAM, המערכת משיגה דיוק ועמידות משופרים במגוון תרחישים.

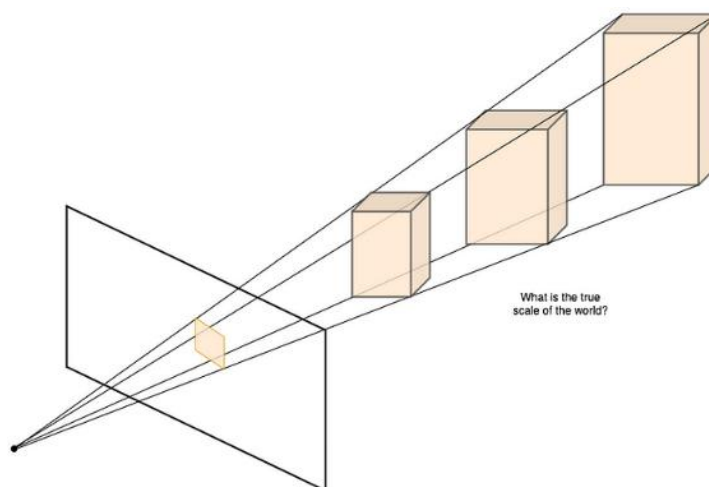
## Abstract

Process a video captured while moving through an environment to output an accurate 3D trajectory of the movement path relative to a known map. By leveraging prior knowledge of the environment and using ArUco markers distributed throughout, we developed a robust system for reconstructing 3D trajectories from video data. The proposed solution employs a hybrid approach combining ArUco marker-based localization and ORB-SLAM3. ArUco markers provide reliable pose estimation, including orientation and location, whenever visible. When markers are not detected, ORB-SLAM3 seamlessly takes over to continue tracking and mapping. By integrating ArUco-derived pose information as ground truth for ORB-SLAM3, the system achieves enhanced accuracy and robustness across diverse scenarios.

# 1. מבוא

מטרת פרק זה היא תיאור קצר של החשיבות והצורך בפרויקט לשחזור מסלולי תנועה תלת-ממדיים מדויקים, והשימוש בטכנולוגיות SLAM [2] ו-ArUco [3].

מערכות חישוביות שמטרתן להבין ולעקוב אחר תנועה בסביבות תלת-ממדיות באמצעות מצלמות. אחת מהטכנולוגיות המרכזיות בתחום זה היא אלגוריתמי SLAM המבוססים על מצלמות מונוקולריות. מקבלים וידאו של סביבה כקלט ומחזירים כפלט מסלול תלת-ממדי של תנועת המצלמה ונקודות עניין במרחב, אך עם זאת, הם סובלים ממגבלות מסוימות שיש להתמודד עמן כדי להבטיח דיוק ועמידות גבוהים יותר. ראשית, חופשיות קנה מידה (Scale Ambiguity): במערכות מונוקולריות, כפי שרואים ב **Error! Reference source not found.**, לא ניתן לקבוע את קנה המידה המוחלט של הסביבה. המשמעות היא שהשחזור התלת-ממדי מדויק רק במונחים יחסיים, ודורש מידע חיצוני כדי לקבוע את הגודל הממשי של אובייקטים במרחב.



איור 1 דוגמה לחופשיות קנה מידה

שנית, SLAM מחזיר מסלול תנועה יחסי בלבד, ללא מיקום מוחלט במרחב הגאוגרפי. ללא ידע מוקדם או נקודות ייחוס חיצוניות, מיקום המערכת נשאר ביחס לנקודת ההתחלה בלבד.

## 2. סקירת ספרות

אלגוריתמי SLAM מונוקולריים :

מערכות SLAM מבוססות מצלמות מונוקולריות כגון ORB-SLAM3 הן בין הכלים הנפוצים ביותר למיפוי ועקיבה בזמן אמת. ORB-SLAM3 משתמש בתכונות חזותיות ייחודיות (כגון ORB features) כדי לעקוב אחר תנועת המצלמה ולבצע מיפוי של נקודות עניין בסביבה. עם זאת, מערכות אלו סובלות ממגבלות כמו חוסר במידע על קנה מידה מוחלט והישענות על תנאי תאורה ואיכות תמונה.

**אפשרויות להתגברות על בעיות ה-SLAM :**

**שימוש במצלמות סטריאו או RGBD :**

מצלמות סטריאו מספקות תמונה תלת-ממדית באמצעות שני חיישנים הממוקמים במרחק ידוע זה מזה. מצלמות RGB-D מוסיפות מידע עומק ישיר באמצעות חיישן עומק ייעודי. פתרונות אלו מספקים מידע על קנה מידה מוחלט אך דורשים מערכות חומרה מתקדמות ויקרות, ולעיתים אינן מתאימות ליישומים ניידים או זולים.

**יחידת מדידה אינרציאלית (IMU) :**

יחידות מדידה אינרציאליות יכולות לספק מידע על דינמיקת התנועה, ולעזור בפתרון בעיות קנה המידה באמצעות שילוב נתוני תאוצה וג'ירוסקופ עם המידע החזותי. עם זאת, פתרון זה דורש חומרה מורכבת ומחייב התאמות במערכת ושיטות לשילוב המידע.

**סמני ArUco :**

סמני ArUco הם סמנים חזותיים שניתן לזהות בקלות באמצעות עיבוד תמונה. הם מספקים מיקום ואוריינטציה מדויקים כאשר הם נראים בתמונה, ובכך משמשים כנקודות ייחוס במערכת ניווט. שימוש בסמנים אלו בשילוב עם SLAM יכול להתגבר על בעיות קנה מידה ומיקום מוחלט, במיוחד בסביבות שבהן קשה לשמור על עקיבה יציבה לאורך זמן.

מטרת סקירת הספרות היא להדגיש את המגבלות של השיטות הקיימות, ולבסס את ההצדקה לשימוש בגישת סמני ArUco המוצעת בפרויקט זה. הגישה משפרת את יכולות המערכת על ידי שילוב מידע ממקורות אמינים ומתן פתרונות לאתגרים מרכזיים כמו קנה מידה ומיקום מוחלט.

## 3. הפתרון

### 3.1. תכנון ראשוני

- בשלב התכנון הראשוני, הוגדרו דרישות המערכת והמטרות המרכזיות:
- יכולת לעבד וידאו מסביבה נתונה ולייצר מסלול תלת-ממדי מדויק.
  - שימוש בשילוב בין SLAM מבוסס מצלמה מונקולרית לבין סמני ArUco.
- במהלך שלב זה נבחרו הכלים המרכזיים לפרויקט:
- **OpenCV** לעיבוד תמונה וזיהוי סמני ArUco.
  - **ORB-SLAM3** לעקיבה ומיפוי רציף.
  - שפת התכנות Python לשם גמישות וקישוריות עם ספריות חיצוניות.

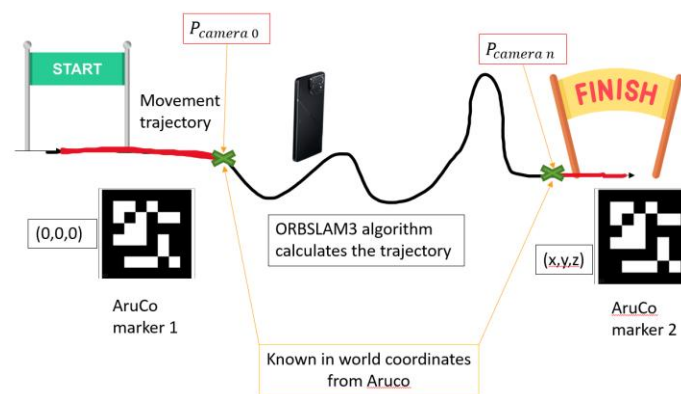
### 3.2. תיאור הפתרון

הפתרון המוצע משלב בין שתי טכנולוגיות עיקריות:

- **זיהוי סמני ArUco**: הסמנים מספקים מיקום מדויק ואוריינטציה כאשר הם נראים בתמונה. מידע זה משמש כנקודת ייחוס למערכת.
- **ORB-SLAM3**: מערכת SLAM עוקבת אחרי תנועת המצלמה וממפה את הסביבה גם כאשר אין סמנים זמינים.

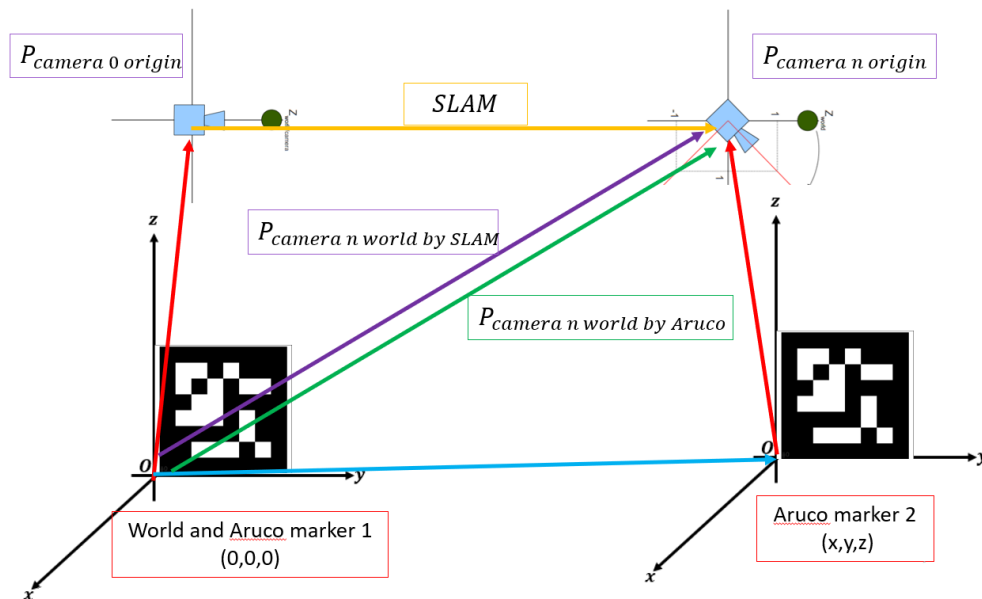
נניח 2 סמנים בסביבה, אחד בתחילת המסלול והשני בסופו, נדע במדויק את מיקומם בעולם. כאשר סמני ArUco נראים, המידע המופק מהם מוזן כקרקע אמת לתוך ORB-SLAM3, מה שמאפשר התגברות על המגבלות. בעת חוסר זיהוי סמנים, ORB-SLAM3 ממשיך לעבוד עצמאית תוך הישענות על המידע הקודם.

באיור 2: ניתן לראות את שלבי המסלול, בקווים האדומים ניתן לראות את אחד הסמנים, במסלול השחור אלגוריתם ה-SLAM ממשיך, האיקסים הירוקים הם הנקודות בהם אנו עוברים בין האלגוריתמים.



איור 2: שלבי המסלול

באיור 3 : כל הווקטורים בבעיה ניתן לראות את 2 הסמנים בתחילת וסוף המסלול, את הווקטור הכחול אנחנו יודעים משום שאנו יודעים איפה במדויק הנחנו את הסמנים, את שני הווקטורים האדומים אנו יודעים כי אנו רואים את אחד הסמנים בכל פעם וכך ניתן לקבל את מיקום המצלמה (חצים אדומים). וקטור צהוב נקבל מה-SLAM, אך הוא יהיה עם חופשיות גודל (scale ambiguity), את הווקטור הירוק והסגול נקבל על ידי חיבור הווקטורים המתאימים. הירוק יהיה גודל אמיתי, הסגול יהיה עם חופשיות גודל, נשווה ביניהם ונחליץ את חופשיות הגודל. משום שאנו יודעים במדויק איפה הסמנים, נדע ישירות איפה המסלול התקיים בעולם האמיתי, וכך התגברנו על שני ההגבלות.



איור 3 : כל הווקטורים בבעיה

### 3.3. תהליך הפיתרון

#### כיול המצלמה

כיול המצלמה הוא שלב קריטי להבטחת דיוק המדידות והמיפוי. תהליך הכיול כלל:

- שימוש בתבנית לוח שחמט (checkerboard) עם ריבועים בגודל ידוע של 2.5 ס"מ.
- צילום תמונות של התבנית מזוויות ומרחקים שונים.
- שימוש באלגוריתמים של OpenCV לחישוב פרמטרי הכיול, כולל מטריצת המצלמה, מקדמי העיוות, ומטריצות סיבוב ותזוזה.
- אימות הכיול על ידי השוואת המידות המחושבות למידות האמיתיות בתמונות הבדיקה.



## פיזור סמני ArUco

על מנת להבטיח זיהוי מדויק של סמני ArUco :

- נניח 2 סמנים, אחד בתחילת המסלול והשני בסופו.
- הסמנים פוזרו במיקומים ידועים מראש במרחקים קבועים.
- גודלם של הסמנים נבחר כך שיתאים לתנאי התאורה ולמרחקי הצילום הצפויים.
- נבדקה יציבות הזיהוי בתנאי תאורה משתנים, כדי להבטיח שימוש אמין.

בעת זיהוי הסמן נפעיל אלגוריתם PnP [1][4] שבסופו נקבל את מטריצת הסיבוב ווקטור ההזזה מהסמן עצמו. נשתמש בהם כדי לקבל את מיקום המצלמה בעת זיהוי הסמן:

Let  $R_{marker}^{camera}$  and  $t_{marker}^{camera}$  be the rotation and translation matrices from world to camera:

$t_{marker}^{camera}$  – 3x1 world origin in camera's coordinates.

$R_{marker}^{camera}$  – 3x3 rotation matrix from world coordinates to the camera's.

$$P_{camera} = R_{marker}^{camera} \cdot P_{marker} + t_{marker}^{camera}$$

$$(R_{marker}^{camera})^{-1} = R_{camera}^{marker} = (R_{marker}^{camera})^T$$

Then we get:  $P_{marker} = (R_{marker}^{camera})^T \cdot (P_{camera} - t_{marker}^{camera})$

For  $P_{camera} = 0$ , we can get the camera origin position in the marker's coordinate:

$$P_{camera\_origin\_in\_marker} = -(R_{marker}^{camera})^T \cdot t_{marker}^{camera}$$

נגדיר את קוארדינטות העולם להיות קוארדינטות של הסמן ונקבל:

$$P_{camera\_origin\_in\_world} = -(R_{world}^{camera})^T \cdot t_{world}^{camera}$$

זהו מיקום המצלמה בעולם, כל עוד אנו רואים את הסמן.

## הרצה של ORB-SLAM3

המודול ORB-SLAM3 הותקן והורץ על מערכת Ubuntu דרך WSL המופעלת ב Windows.

תהליך ההרצה כלל:

- התקנת כל התלויות הנדרשות, כולל Pangolin וספריות גרפיקה.
- התאמת קובץ ההגדרות של המצלמה, תוך שימוש בפרמטרים מהכיל שנעשה.
- הפעלת האלגוריתם על סרטונים מוקלטים ובדיקת תוצאות המיפוי והעקיבה בתרחישים שונים.

הפעלה זו אפשרה גישה ל ORB-SLAM3 תוך ניצול סביבת עבודה נוחה ב Windows.

ORB-SLAM3 gives us the transformation matrix between camera 0 and camera i.

This is a new coordinate system! camera 0 is at (0,0,0) here.

Let  $R_{camera\_i}^{camera\_0}$  and  $t_{camera\_i}^{camera\_0}$  be the rotation and translation matrices from camera i to camera 0.

$$P_{camera\_0} = R_{camera\_i}^{camera\_0} \cdot P_{camera\_i} + \alpha \cdot t_{camera\_i}^{camera\_0}, \alpha \text{ is the scale ambiguity problem.}$$

## חיבור בין המודולים

אינטגרציה בין רכיבי המערכת בוצעה כך:

1. זיהוי סמני ArUco מתבצע על כל פריים שבו הם גלויים.
2. לאחר הזיהוי מתבצע חישוב מיקום המצלמה הנוכחי לפי אלגוריתם PnP.
3. במידה ואין סמנים בפריים, ORB-SLAM3 ממשיך לנתח את המסלול בהתבסס על המידע שנצבר.
4. ORB-SLAM3:

$$P_{camera\ 0} = R_{camera_i}^{camera_0} \cdot P_{camera\ i} + \alpha \cdot t_{camera_i}^{camera_0}$$

עבור:

$$i=n, P_{camera\ n} = 0$$

$$P_{camera\_n\_origin\_in\_camera\_0} = \alpha \cdot t_{camera_n}^{camera_0}$$

נזכור מ ArUco על camera 0 נקבל:

$$P_{world} = (R_{world}^{camera})^T \cdot (P_{camera} - t_{world}^{camera})$$

נציב את camera n לפי קוארדינטות camera 0 כדי לקבל את Camera n בקוארדינטות העולם:

$$P_{camera\_n\_in\_world\_by\_SLAM} = R_{0\_aruco}^T \cdot (\alpha \cdot t_{camera_n}^{camera_0} - t_{0\_aruco})$$

מ ArUco על camera n נקבל:

$$P_{camera\_n\_origin\_world\_location\_by\_aruco} = -R_{world}^{aruco_n T} t_{world}^{aruco_n} + P_{marker\ 2}$$

נשווה ביניהם ונמצא את  $\alpha$ :

$$R_{0\_aruco}^T \cdot (\alpha \cdot t_{camera_n}^{camera_0} - t_{0\_aruco}) = -R_{n\_aruco}^T t_{n\_aruco} + P_{marker\ 2}$$

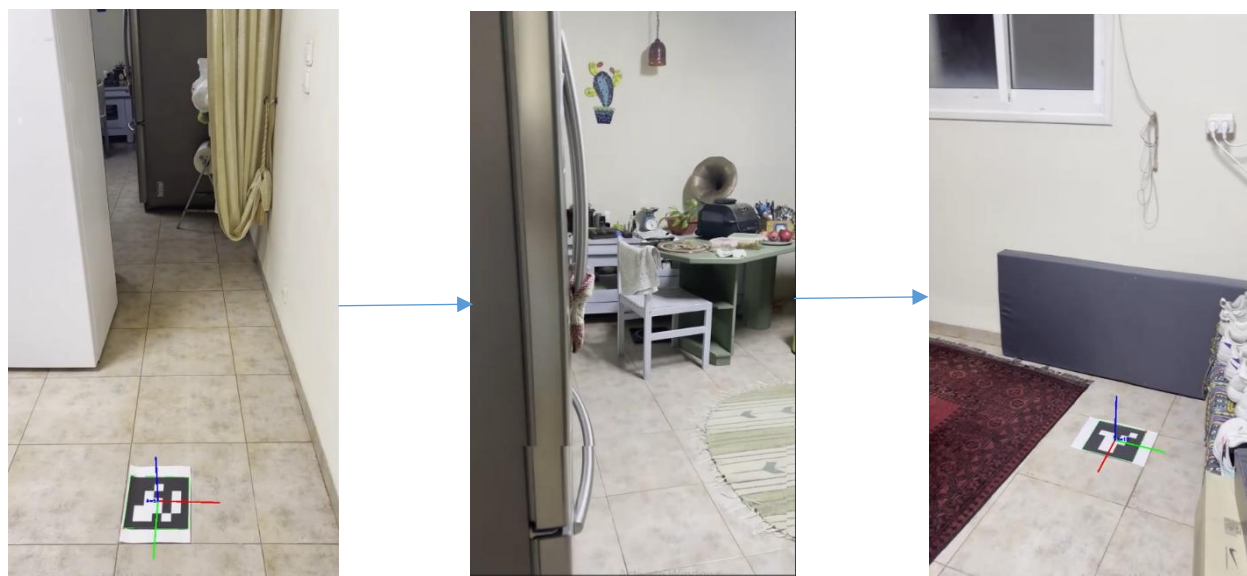
$$\alpha = \frac{(-R_{0\_aruco} R_{n\_aruco}^T T_{n\_aruco} + R_{0\_aruco} P_{marker\ 2} + T_{0\_aruco})}{T_{n\_SLAM}}$$

5. מנגנון ניהול מבטיח שהמידע משולב בצורה אופטימלית ללא חריגות או סטיות.

### 3.4. תוצאות

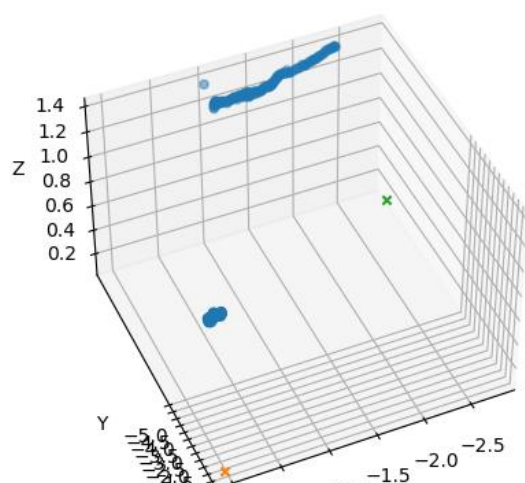
המערכת נבדקה בתרחישים שונים כדי להעריך את ביצועיה:

- באיור 4: משמאל לימין, מסלול ראשון, עשיר פיצצ'רים תרחיש עם הרבה פיצצ'רים ייחודיים, הושג דיוק גבוה של המסלול, כולל התאמה טובה לקנה המידה.



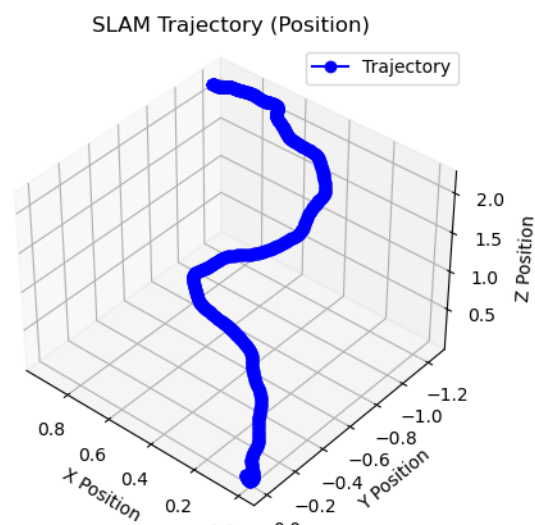
איור 4: משמאל לימין, מסלול ראשון, עשיר פיצצ'רים

ArUco:



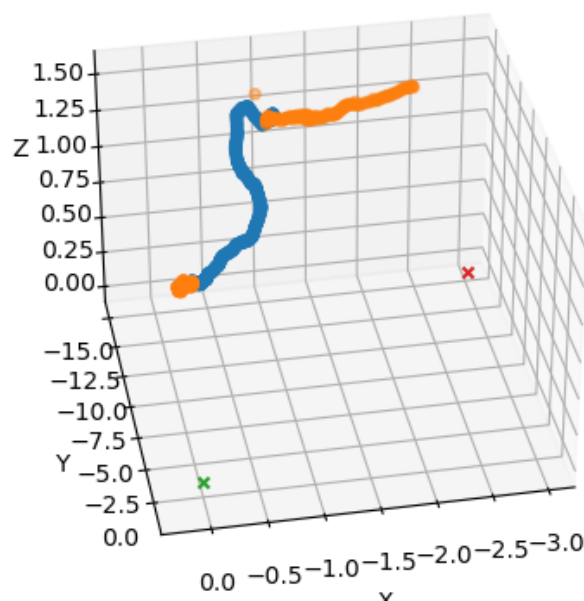
איור 6 : פלט תהליך זיהוי ומציאת מיקום מצלמה לפי הסמנים. קיים רק בתחילת וסוף מסלול (איפה שרואים את הסמנים).

ORB-SLAM3:



איור 5 : פלט ORB-SLAM3, מסלול יחסי בלבד ובלי גודל אמיתי.

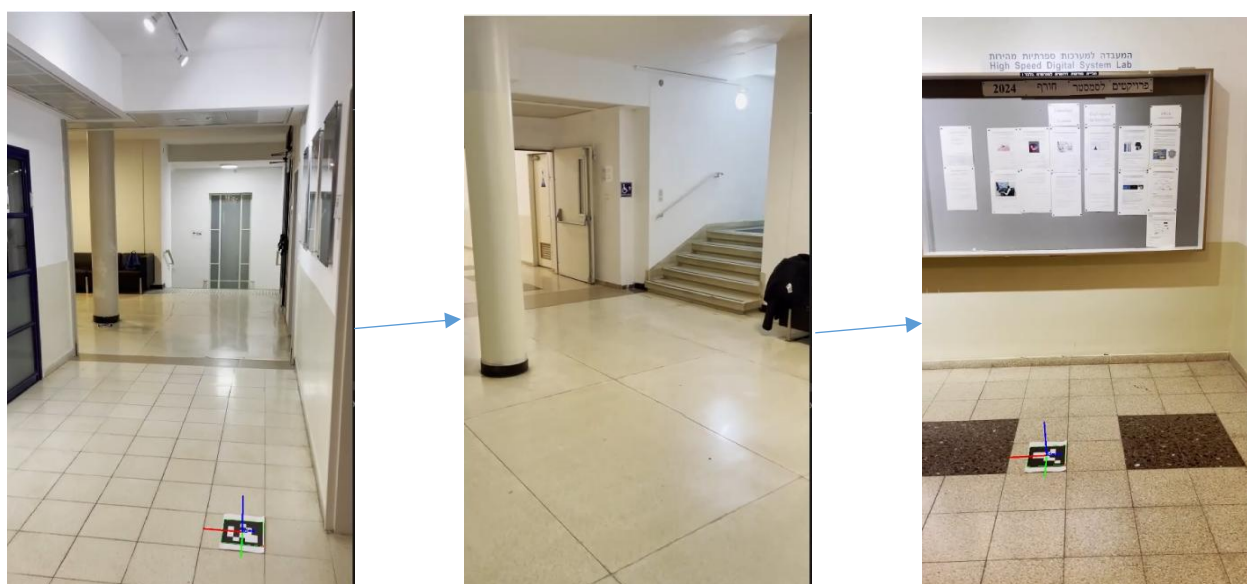
באיור 7 : פלט סופי, המסלול בגודל אמיתי עם התייחסות למפה (איקסים הם הסמנים), הפלט לאחר חיבור



איור 7 : פלט סופי, המסלול בגודל אמיתי עם התייחסות למפה (איקסים הם הסמנים).

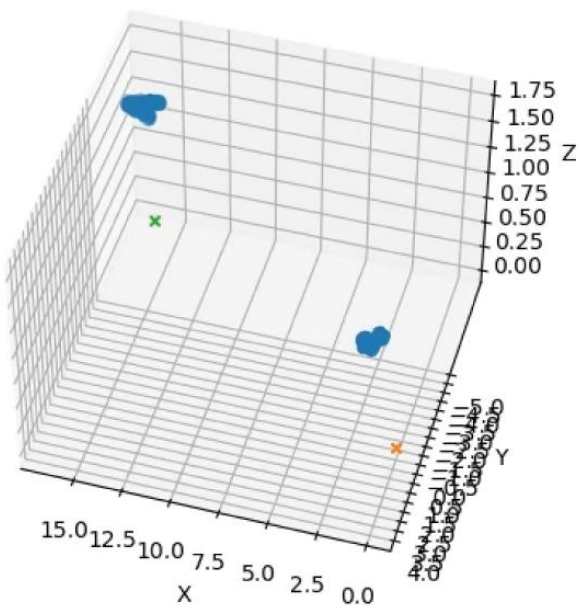
מודלים:

- באיור 8: משמאל לימין, מסלול שני, דל פיצ'רים תרחיש של פיצ'רים ייחודיים: הושג דיוק טוב של המסלול, כולל התאמה טובה לקנה המידה.

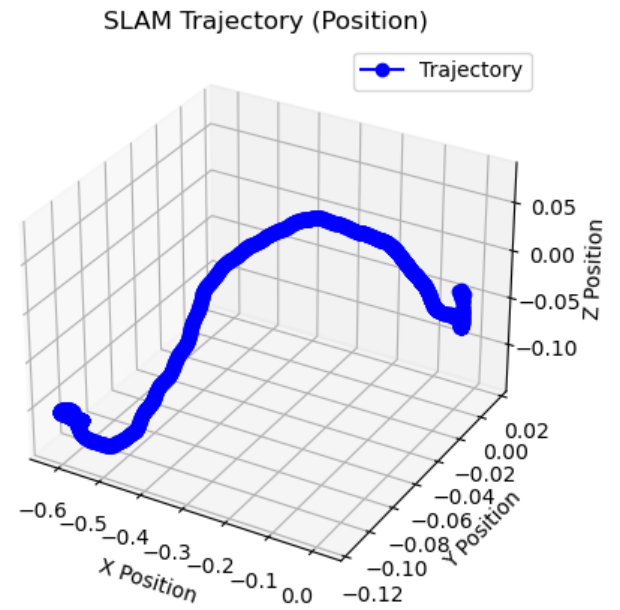


איור 8: משמאל לימין, מסלול שני, דל פיצ'רים

ArUco:

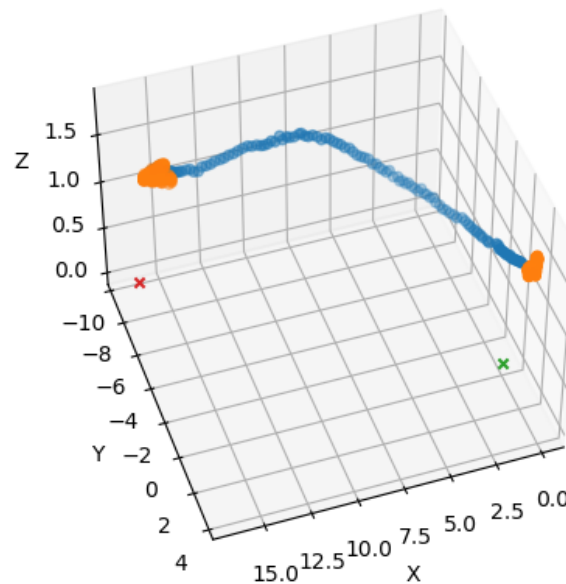


ORB-SLAM3:



איור 10: פלט ORB-SLAM3, מסלול יחסי בלבד ובלי גודל אמיתי.

לאחר חיבור מודלים:



איור 11: פלט סופי, המסלול בגודל אמיתי עם התייחסות למפה (איקסים הם הסמנים).

תוצאות הבדיקות מראות כי הפתרון ההיברידי מתגבר על מגבלות ה-ORB-SLAM3 בצורה נוחה יחסית ומחזיר מפת מסלול 3D בדיוק טוב.

## 4. סיכום

הפרויקט עוסק בפיתוח מערכת מתקדמת לשחזור מסלול תלת-ממדי על בסיס נתוני וידאו שצולמו במהלך תנועה בסביבה. המערכת משלבת בין סמני ArUco להפקת מידע מדויק על מיקום ואוריינטציה, לבין ORB-SLAM3, אלגוריתם SLAM מונוקולרי, לצורך מעקב ומיפוי רציף של הסביבה. הפתרון ההיברידי מאפשר למערכת להתמודד עם מגבלות שונות של SLAM מונוקולרי, כגון חוסר בקנה מידה מוחלט וחוסר יחוס למפה האמיתית. סמני ArUco משמשים כנקודות ייחוס שמספקות מידע מדויק בעת זיהויים, בעוד ש-ORB-SLAM3 מבצע מעקב ומיפוי כאשר הסמנים אינם נראים במצלמה. במהלך העבודה:

- בוצע כיוול מצלמה.
- פותח מנגנון לקליטת מידע מסמנים והמרתו למיקום אמיתי של המצלמה.
- פותח מנגנון לשילוב מידע מסמנים ומ-ORB-SLAM3 תוך התמודדות עם בעיית קנה המידה.

המערכת נבדקה בתרחישים שונים, כולל סביבות עם פיצ'רים ייחודיים ותנאי תאורה משתנים. הבדיקות מראות כי הפתרון ההיברידי מתגבר על מגבלות ה-ORB-SLAM3 בצורה נוחה יחסית ומחזיר מפת מסלול 3D בדיוק טוב.

לסיכום, הפתרון המוצע מדגים כיצד שילוב טכנולוגיות SLAM וסמנים חזותיים יכול להוביל לשחזור מסלולים תלת-ממדיים מדויקים, ומהווה פתרון נוח וטוב בתחום המיפוי והמעקב.

## רשימת מקורות

- [1] " PnP with ArUco.," 2015. [Online]. Available:  
[https://docs.opencv.org/4.x/d5/dae/tutorial\\_aruco\\_detection.html](https://docs.opencv.org/4.x/d5/dae/tutorial_aruco_detection.html).
- [2] Carlos Campos, et al, "ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual-Inertial and Multi-Map SLAM." IEEE Transactions on Robotics 37.6, 2021
- [3] "ArUco Markers" [Online]. Available: <https://github.com/PlusToolkit/aruco>
- [4] PnP [https://docs.opencv.org/4.x/d5/d1f/calib3d\\_solvePnP.html](https://docs.opencv.org/4.x/d5/d1f/calib3d_solvePnP.html)