



AI : רכב רובוטי חכם מבוסס MARICH

מוגש ע"י :

מריה נחלה – 207716762

כריסטיאן חמיסה – 324304955

בהנחיית : אלחנדרו גוליוחוב

 חתימה :

29.9.2025 ז' בתשרי ה'תשפ"ו

הוגש לשם מילוי חלקי של הדרישות לקבלת תואר
"בוגר במדעים B.Sc. בהנדסת חשמל ואלקטרוניקה"

0. תקציר:

הפרויקט בוצע במסגרת תכן הנדסי פנימי פרויקט גמר ב"מכללת אורט בראודה" כרמיאל.

במסגרת פרויקט גמר זה פותח רכב רובוטי חכם מבוסס Raspberry Pi 5 המשולב עם מסך מגע קיבולי בגודל 7 אינץ', מערכת ראייה ממוחשבת ובנייה מלאכותית מתקדמת, המאפשרת לו לזהות רגשות, מחוות ותנועות גוף, לבצע מעקב פנים, ולזהות תווי פנים.

הרכב מבוסס על צ'אסיס מתכתי מצויד בגלגלי Mecanum המאפשרים תנועה 360° ומתרגם ליכולת ניווט בכיוונים שונים. החומרה כוללת מצלמת USB בגובה MP 1 עם 2-DOF PTZ, חיישני אולטרסוניק, חיישני מעקב (4-ערוצים), פסי RGB, תצוגת OLED ובומר (buzzer).

מבחינת תוכנה, נעשה שימוש בספריות OpenCV ו-MediaPipe ליישומי זיהוי צבע, מעקב מטרות, זיהוי לוחיות רישוי, מעקב ופענוח פנים, זיהוי מחוות ותנועות גוף.

סביבת הפיתוח מבוססת Python עם התממשקות דרך :

ROS2 (Humble Edition) המאפשרת למידה הדרגתית, שליטה נוחה ותשתית מתקדמת לרובוטיקה חינוכית והתנסותית.

המטרה המרכזית היא ליצור מערכת אינטראקטיבית טבעית ואינטואיטיבית, המסוגלת להגיב בזמן אמת לסביבה האנושית עם ממשק מגע, תגובות ויזואליות ובסיס למידה וגמישות. הפרויקט משמש כפלטפורמה חינוכית המיועדת ללימוד רובוטיקה ו-AI המותאמת במיוחד לסטודנטים ולחובבים.

2	0. תקציר:
6	1. מבוא:
6	2. תיאור המערכת:
6	2.1 חומרה
7	2.2 תוכנה
7	2.3 מפרט פונקציונלי:
7	2.3.1 מערכת לבישה (On-board System)
8	2.3.2 מרכז הבקרה (Control Center)
8	2.4 מפרט טכני – טבלת פירוט מכלולי המערכת
9	2.5 תרשים מלבנים:
10	2.6 עקרון פעולת המערכת:
10	3. מטלות:
10	3.1 מטלות הנדסיות:
11	3.2 ביצוע המטלות ע"י המתמחה :
11	3.2.1 שלבי התכנון :
16	3.2.2 תכנון אב
19	3.2.3 תרשימי זרימה
22	3.2.4 סכמה חשמלית
23	3.2.5 סימולציות
30	3.3 בעיות ואתגרים הנדסיים
31	4. סיכום ודיון :
31	4.1 הצורך בפרויקט :
32	4.2 סיכום ומסקנות:
32	4.3 תוספות עתידיות:
33	4.4 עמידה בדרישות:
34	5. ביבליוגרפיה:
34	6. נספחים:

רשימתה טבלאות וגרפים:

9.....	טבלה 1: מפרט תכני
29.....	גרף 1: זמני התגובה של ה-AI
29.....	טבלה 2: דיוק התשובות
33.....	טבלה 3: עמידה בדרישות

רשימת האיורים:

9.....	איור 1: תרשים מלבינים
13.....	איור 2: הרכבת המכונית, שלב 1
13.....	איור 3: הרכבת המכונית, שלב 2
14.....	איור 4: הרכבת המכונית, שלב 3
14.....	איור 5: הרכבת המכונית, שלב 4
15.....	איור 6: מסך LCD
15.....	איור 7: רכב מלא ללא מסך
15.....	איור 8: התוצאה הסופית
19.....	איור 9: תרשים זרימה כללי
20.....	איור 10: תרשים זרימה IR
21.....	איור 11: תרשים זרימה AI
22.....	איור 12: סכמה חשמלית
23.....	איור 13: זיהוי פנים באור חזק
23.....	איור 14: זיהוי פנים באור בינוני
24.....	איור 15: זיהוי פנים באור נמוך
24.....	איור 16: זיהוי פנים כאשר מספר פרצופים זמינים
25.....	איור 17: תנועת יד עוקבת
26.....	איור 18: זיהוי צבע
27.....	איור 19: זיהוי גוונים שונים של כחול
28.....	איור 20: זיהוי אובייקטים
28.....	איור 21: זיהוי לוחות רישוי

AI – Artificial Intelligence

B.Sc. – Bachelor of Science

CPU – Central Processing Unit

CSV – Comma Separated Values

DOF – Degrees of Freedom

FPV – First Person View

GUI – Graphical User Interface

I2C – Inter-Integrated Circuit

IP – Internet Protocol

IR – Infrared

LCD – Liquid Crystal Display

LLM – Large Language Model

MP – Megapixel

OLED – Organic Light-Emitting Diode

PTZ – Pan, Tilt, Zoom (Camera movement)

RGB – Red, Green, Blue

ROS2 – Robot Operating System 2

STT – Speech-to-Text

TCP – Transmission Control Protocol

TTS – Text-to-Speech

USB – Universal Serial Bus

Wi-Fi – Wireless Fidelity

1. מבוא:

שילוב הרובוטיקה עם בינה מלאכותית וראייה ממוחשבת מציב אתגרים משמעותיים, ולא פחות מכך – פוטנציאל לשיפור איכות החיים והלימוד בתחום. כיום, רובוטים אינטראקטיביים מעטים משולבים עם ממשקים טבעיים כמו מסכי מגע, יכולות זיהוי רגשות ותגובה בזמן אמת, במיוחד בפיתוחים חינוכיים שיכולים להנגיש טכנולוגיות מתקדמות לסטודנטים וחובבים.

פרויקט זה מתמקד בתכנון ובנייה של **רכב רובוטי חכם** מבוסס **Raspberry Pi 5** המשולב במסך מגע בקיבולי בגודל **7 אינץ'**, עם **יכולת תנועה רב-כיוונית** באמצעות גלגלי **Mecanum** וזיהוי מתקדם של מחוות, רגשות, פנים ומדידת מרחק בעזרת מצלמת USB עם **PTZ-דו-ממדי**, חיישני אולטרסוניק, פסים מוארים, (RGB) ותצוגת OLED. כל זאת, תוך שילוב טכנולוגיות כמו **OpenCV**, **MediaPipe** ו-**ROS2** ב-Python לשם יצירת אינטראקציה טבעית ותגובה מהירה עם הסביבה האנושית.

המטרה המרכזית היא לפתח מערכת אינטואיטיבית, אינטראקטיבית וגמישה, המספקת ממשק מגע וראייה לרובוט, ומאפשרת לו לקבל החלטות ולהגיב בזמן אמת. פרט לכך, הפרויקט פותח כפלטפורמה **חינוכית ומעשית**, המאפשרת למידה הדרגתית של טכנולוגיות רובוטיקה, ראייה ממוחשבת ו-AI-תוך מתן אפשרות להרחבה ושדרוג עתידי קל.

2. תיאור המערכת:

2.1 חומרה

- בקר מרכזי **Raspberry Pi 5**: מארח סביבת **ROS2 (Humble Edition)** ומתכנת ב-Python פתרון מתאים גם למתחילים וגם למתקדמים.
- שלדה: גוף מתכתי עמיד התומך בגלגלי **Mecanum**, המאפשרים תנועה חלקה בכל כיוון (כולל צדדים וסיבובים במקום).
- מסך מגע קיבולי בגודל **7" (7" LCD)**: משולב לממשק משתמש ויזואלי אינטואיטיבי, מציג מידע בזמן אמת ומאפשר שליטה ישירה ללא צורך לשימוש במחשב חיצוני – תורם לחוויית משתמש חלקה, הופך את הרובוט לפלטפורמה עצמאית ונגישה.
- מצלמה חכמה: מצלמת USB ברזולוציה בינונית (**1MP**) המותקנת על גימבל **PTZ (2-DOF)** מאפשרת תנועות מעל ומתחת ומסביב לצורך מעקב ותצוגה חכמה.
- חיישנים משלימים: כוללים חיישני עקיבה (**4-ערוצים**), חיישן אולטרסוניק למדידת מרחקים, פסי תאורת **RGB**, תצוגת **OLED** אינפורמטיבית ו-buzzer לצלילים – כולם מחוברים דרך הרחבה ייעודית.

2.2 תוכנה

- ספריות ואלגוריתמים : משתמשים ב-OpenCV לעיבוד תמונה ו-MediaPipe לזיהוי תנועות גוף, פרצופים ומחוות.
- מערכת ROS2 : מאפשרת ארכיטקטורה מודולרית וקלה לפיתוח, עם תמיכה ב-rviz2, rqt ו-ROS micro אידאלית ללמידה, לתחרות ולמחקר.
- שליטה מרובת ערוצים : פיקוד נעשה דרך אפליקציית FPV חיבור דרך (Wi-Fi), שלט אינפרא-אדום או דרך מחשב עם סביבת ROS מבטיח גמישות שימוש.

2.3 מפרט פונקציונלי :

2.3.1 מערכת לבישה (On-board System)

1. **יכולת קליטת נתוני תנועה**
המערכת כוללת חיישני עקיבה (4-ערוצים) ומצלמה עם PTZ המאפשרים קליטת תנועות, קווים ומעקב חזותי.
2. **יכולת קליטת מצב תאורה**
מערכת OpenCV מאפשרת המרת תמונת הסביבה לזיהוי צבעים ותנאי תאורה באמצעות מסך ומצלמה.
3. **יכולת צילום**
מצלמת USB ברזולוציה ~MP 1 עם גימבל 2-DOF מבצעת צילום ועיבוד בזמן אמת.
4. **יכולת עיבוד נתונים וקבלת החלטות**
שליטת המערכת נעשית באמצעות Raspberry Pi 5 בשילוב ROS2 ו-Python, תוך שימוש ב-OpenCV ו-MediaPipe לזיהוי מחוות, פרצופים, תווי רכב ועוד.
5. **יכולת עבודה רצופה באופן אלחוטי למשך שלוש שעות**
אמנם לא צוין במפורש, אך המערכת פועלת באמצעות סוללת V 7.4 ליתיום סולידית, שמספקת פעולה ממושכת ללא חוטי חיבור.
6. **יכולת התראה בזמן אמת לנמען מוגדר**
כוללת פלט חישה חזותי (RGB, OLED) וקולי (buzzer) כאמצעי התראה מיידי.
7. **יכולת העברת נתונים (תמונות + דאטה) באמצעות פרוטוקול IP-TCP**
תמיכת ROS2 ותקשורת דרך Wi-Fi מאפשרים שידור נתונים בזמן אמת למכשירים חיצוניים.
8. **יכולת עבודה באופן אוטונומי**
המערכת תומכת בעקיבה עצמונית (line tracking, visual tracking), הימנעות ממכשולים וזיהוי תווים, QR, תווי פנים באופן אוטונומי.

2.3.2 מרכז הבקרה: (Control Center)

9. יכולת עיבוד נתונים וקבלת החלטות

מרכז הבקר (ה-Raspberry Pi 5) מנהל שליטה חכמה, מעבד נתוני מצלמה וחיישנים ומחליט על תגובות.

10. יכולת קבלת נתונים באמצעות פרוטוקול IP-TCP

מאפשר קבלת נתונים בזמן אמת מהחיישנים דרך Wi-Fi או ROS2.

11. יכולת שמירת נתונים (לצורך מעקב ולמידת הרגלים)

אפשר לשמור תמונות, לוגים ונתונים לניתוח מאוחר או אימון מודלים ROS2 – תומך בלוגים ושמירת מידע.

12. יכולת צילום

גם במרכז הבקרה נעשה תיעוד רגיל של תמונות/וידאו באמצעות אותה מצלמה, לשם ניתוח או תצוגה בזמן אמת.

13. יכולת התראה בזמן אמת

תצוגה חזותית (על מסך 7"), פלט RGB/OLED וב-Buzzer מאפשרים התראה והתראה באמצעות מרכז הבקרה.

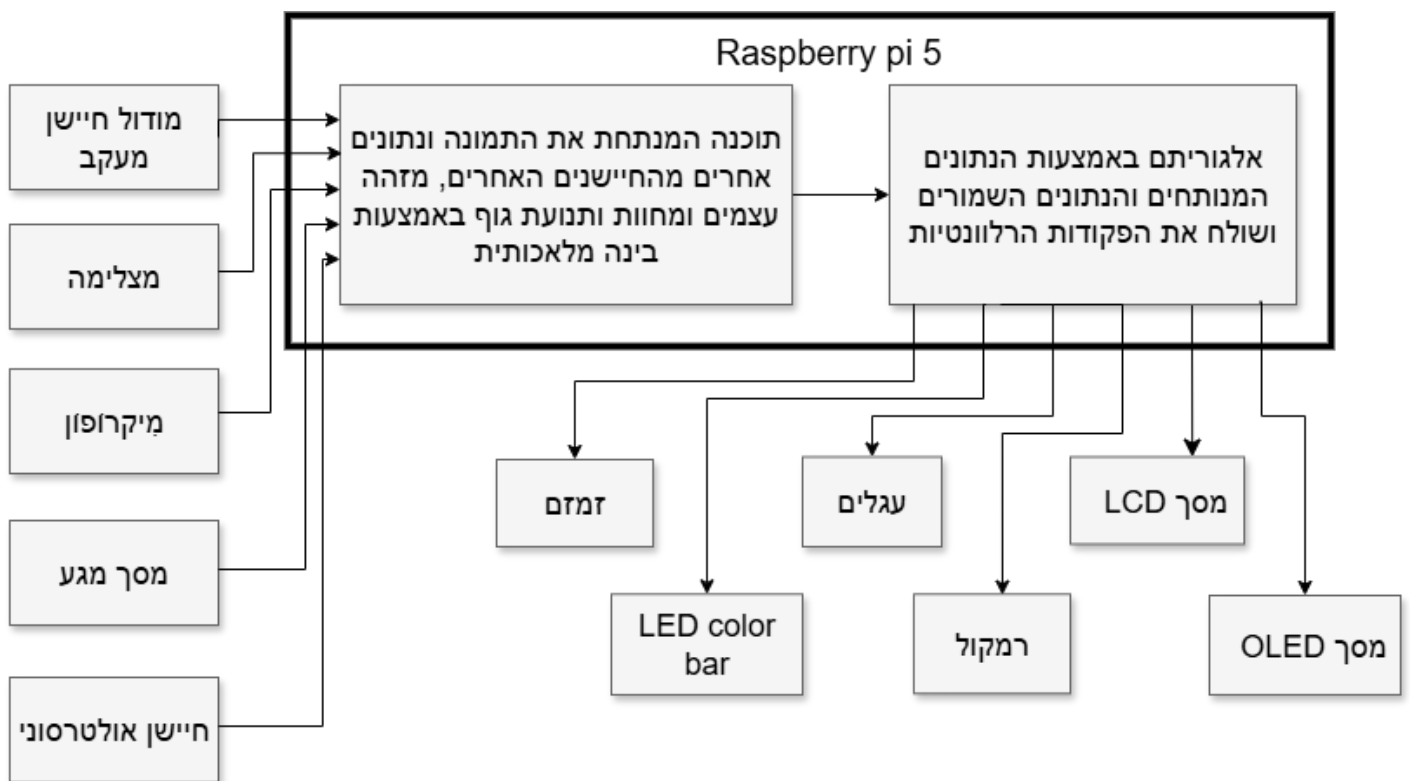
2.4 מפרט טכני – טבלת פירוט מכלולי המערכת

מספר	רכיב / מכלול	תיאור תכליתי	מפרטים טכניים עיקריים
1	בקר ראשי : Raspberry Pi 5	איסוף נתונים מהחיישנים, עיבודם, קבלת החלטות, שליטה בתנועה ובתגובות כלפי ממשק המשתמש ומודולי AI	ROS2 Humble, Python, I2C, USB-C PD 5V/5A
2	מסך מגע קיבולי 7" (LCD)	ממשק משתמש ישיר לצפייה במצב הרובוט, זיהויים ותגובות; מאפשר שליטה ללא תלות במחשב חיצוני	ביטוי ויזואלי, מגע רב-תכליתי (עדויות מהפרויקט)
3	מצלמת USB 1 MP + PTZ 2-DOF	צילום וסריקה בזמן אמת של הסביבה — עקיבה אחרי פנים, מחוות, תווי לוחיות; תגובה בזמן אמת באמצעות עיבוד AI	רזולוציה 480 FoV, p/720p, 120° דיאגונלי, סריקה אופקי + אנכי
4	גלגלי Mekanum + מנועים	תנועה בכל כיוון — קדימה, אחורה, הצדה, אלכסון וסיבוב במקום, כולל בתנאים מורכבים	מוטור, ~245 RPM TT, מומנט 0.8 Nm
5	חיישני עקיבה (4-ערוצים)	זיהוי מסלול רצוף (line tracking) ופיקוד על התנועה בהתאם לקרקע	חיישני אינפרא-אדום מדויקים במיוחד
6	חיישן אולטרסוניק	מידת מרחקים והימנעות ממכשולים באופן אוטונומי	מודול קצה קדמי, זיהוי עצמים לאורך הדרך

7	פלטס OLED : תצוגה + RGB , Buzzer	הצגת נתונים (IP, CPU) מצב מערכת, זיהוי AI, אינדיקציה חזותית וקולית מהירה למשתמש	OLED להצגת סטטוס ; RGB בר אלקטרוני עם buzzer, אפקטים
8	כלי שליטה חיצוניים	שמאפשרים שליטה באופנים מגוונים : אפליקציית FPV שלט IR או ממשק מחשב דרך ROS2	Wi-Fi Hotspot FPV , שלט אינפרא-אדום, ממשק PC
9	אספקת אנרגיה : סוללת 7.4 V 2S Lipo	מאפשרת פעולה רציפה ללא תלות במתח קווי, עבודה בשטח ובזמן אמת	2000 mAh, תצפיות עבודת 2.1 שעות
10	לוח הרחבה (Expansion Board)	ממשק בין Raspberry Pi לחיישנים, מנועים ורכיבים אקטואטיביים אחרים	כולל חיבורים ל I2C- מנועים, OLED, חיישנים, משדר IR, דחפים וכפתורים

טבלה 1: מפרט תכני

2.5 תרשים מלבנים :



איור 1: תרשים מלבנים

2.6 עקרון פעולת המערכת:

המערכת מבוססת על יחידת עיבוד ראשית Raspberry Pi 5 המחוברת למצלמת USB באיכות MP 1 המותקנת על מתקן PTZ דו-צירי (2DOF) ולמסך מגע קיבולי בגודל 7 אינץ'. בעת הפעלת המערכת, הרובוט מתחבר לסביבתו ומתחיל לאסוף נתונים מהמצלמה ומהחיישנים השונים (אולטרסוניק, חיישני מעקב, סרגל תאורה RGB). המידע שנאסף עובר עיבוד בזמן אמת באמצעות ספריית OpenCV ומסגרת MediaPipe לזיהוי פנים, מעקב אחר תנועות, זיהוי צבעים, מחוות ידיים ולוחות רישוי. במקביל, תצוגת הרובוט במסך המגע מדמה הבעות פנים ורגשות בהתאם לסיטואציה באמצעות ממשק גרפי (GUI) אינטואיטיבי. מערכת החישה והעיבוד מתואמת עם מנועי גלגלי ה-Mecanum המאפשרים תנועה חכמה וחלקה לכל כיוון, תוך הימנעות ממכשולים בעזרת חיישני האולטרסוניק. המערכת משלבת בין עיבוד נתונים, תגובה מיידית ותנועה מתקדמת ליצירת אינטראקציה טבעית וחכמה עם המשתמשים והסביבה.

3. מטלות:

3.1 מטלות הנדסיות:

- **תכנון ובניית המערכת המכאנית** – הרכבת שלדת מתכת, התקנת גלגלי Mecanum ומנועים לתנועה חכמה.
הערכת זמן: שבועיים
- **חיבור והטמעת רכיבי החומרה** – שילוב מצלמת USB על מתקן PTZ, חיישני אולטרסוניק, מסך מגע קיבולי, חיישני מעקב OLED, ותאורת RGB.
הערכת זמן: שבוע
- **פיתוח אלגוריתמים לעיבוד תמונה וזיהוי אובייקטים** – יישום OpenCV ו-MediaPipe לזיהוי פנים, מעקב תנועות, זיהוי צבעים ומחוות.
הערכת זמן: שלושה שבועות
- **פיתוח ממשק גרפי (GUI)** – עיצוב ובניית תצוגת רגשות ודמות אינטראקטיבית על מסך המגע.
הערכת זמן: שבועיים
- **פיתוח מערכת בקרה ותנועה** – כתיבת קוד לשליטה במנועים ובגלגלים, תיאום תנועה חלקה בעזרת חיישני אולטרסוניק.
הערכת זמן: שבועיים
- **אינטגרציה בין רכיבי חומרה לתוכנה** – חיבור מלא בין המערכת המכאנית, חיישנים, אלגוריתמים וממשק המשתמש.
הערכת זמן: שבועיים

- **בדיקות מערכת ואופטימיזציה** – הרצת ניסויים, איתור תקלות ושיפור ביצועים למערכת יציבה ויעילה.
הערכת זמן : שבועיים
- **הכנת דוח מסכם והצגת הפרויקט** – כתיבת דוח טכני, הכנת מצגת והדגמת המערכת.
הערכת זמן : שבוע
- **סה"כ משך זמן מוערך: כ-12 שבועות.**

3.2 ביצוע המטלות ע"י המתמחה :

3.2.1 שלבי התכנון :

שלבי התכנון וביצוע המטלות

המשימות בוצעו בהתאם לארבעה תחומים עיקריים, עם הפרדה ברורה בין תכנון חומרה, תוכנה, תקשורת ואינטראקציה חכמה :

א. פיתוח ממשק IP להעברת נתונים

מטרה: ליצור ערוץ תקשורת יציב להעברת תמונות, קבצי לוג או CSV בין מערכת הלבשה למרכז הבקרה (או לסביבה חיצונית).
ביצוע:

- שימוש ב-ROS2 וב-Wi-Fi לצורך תקשורת TCP/IP בין רכיבי המערכת.
- כתיבת מודול Python (Publisher/Subscriber), המאפשר שליחת קבצים (תמונות, לוגים) בזמן אמת.
- בדיקה עם משלוח ומעקב של קבצים בין הלוח לרובוט, תוך חייווי חזותי על שידור מוצלח.
תוצאה: ממשק תקשורת אמין לשליחת נתונים מקומי ובין-מכונה.

ב. פיתוח מודול AI לזיהוי מחוות, פרצופים ותנועות גוף

מטרה: להעניק לרובוט יכולת אינטראקציה טבעית עם משתמש, בזיהוי מחוות ותנועות גוף.
ביצוע:

- שימוש ב-OpenCV וב-MediaPipe לזיהוי תווים חזותיים (כולל פרצופים ומחוות גוף) בזמן אמת.
- עריכת ניסויים: ביצוע תרחישי תנועה שונים (הצגת פנים, הרמת יד, תנועה מקוונת) כדי לכייל Thresholds ורגישות.

- הטמעת תגובה מהירה לדוגמה: שינוי בצבע פס RGB או תצוגת טקסט/קול בהתאם למחווה מזוהה.
- תוצאה:** אלגוריתם יציב שמזהה מחוות תוך זמן תגובה מהיר ומונחה זיהוי אמין בסביבה מבוקרת.

ג. תכנון חומרה ומיקום רכיבים במערכת הלבשה

מטרה: לארגן פיזית את הרכיבים על הגוף הרובוטי לחוויית אינטראקציה חלקה ותפקוד רציף בשטח.

ביצוע:

- קביעת מיקום מצלמת ה-USB על גימבל PTZ לזווית ראייה מקסימלית.
- סידור חיישני IR, אולטרסוניק, פס RGB, OLED, ו-buzzer באופן שמאפשר קלות גישה ונקודת חיבור נוחה ללוח ההרחבה.
- בחירה בסוללת ליתיום (7.4 V, 2000 mAh) הכוללת איזון משקל (פחות מ-100 גרם) ומספקת זמן עבודה ממוצע של כ-3 שעות.
- תוצאה:** מערכת יציבה, מאוזנת ומשקל נמוך — מתאימה לשימוש תנועתי ולא דורשת חיבור חיצוני במהלך פעילות.

ד. תכנון מרכז הבקרה ותפקודיו

מטרה: ליצור מודול בקרה חכם שמרכז את עיבוד הנתונים, השמירה והתגובה בזמן אמת.

ביצוע:

- פיתוח מודול ROS2 ב-Python לתחקור אירועים, שמירת לוגים ותיעוד תמונות.
- אינטגרציה עם רכיבים: תצוגת OLED להצגת IP, מצב מערכת, מצב AI; פס RGB להצגת סטטוס ו-buzzer-לאזהרות קוליות.
- תכנון מנגנון תגובה — לדוגמה: בעת זיהוי מחווה, שינוי צבע הפס והצגת הודעה על המסך.
- תוצאה:** מרכז בקרה שמנטר את הפעילות, מתעד ויכול להתריע בזמן אמת על אירועים.

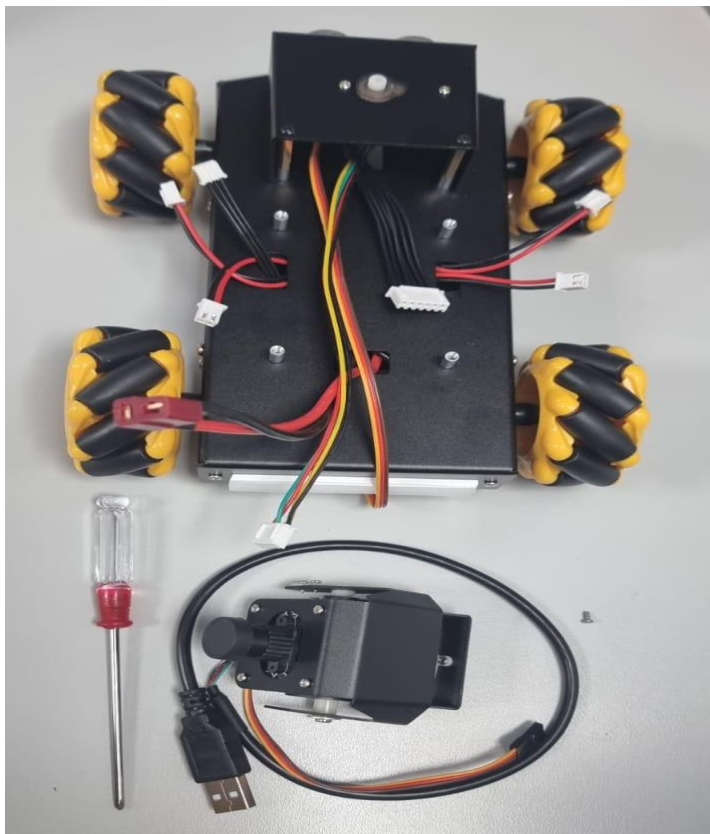
איך התחלנו



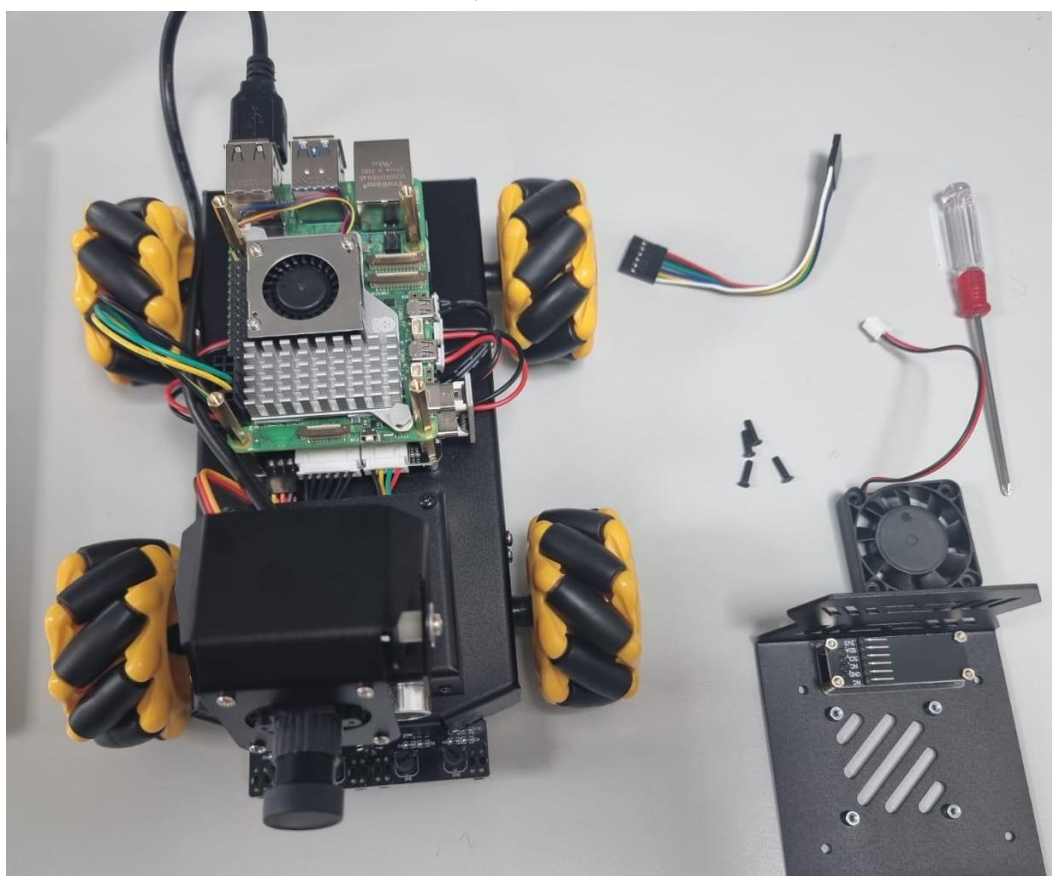
איור 2: הרכבת המכונית, שלב 1



איור 3: הרכבת המכונית, שלב 2



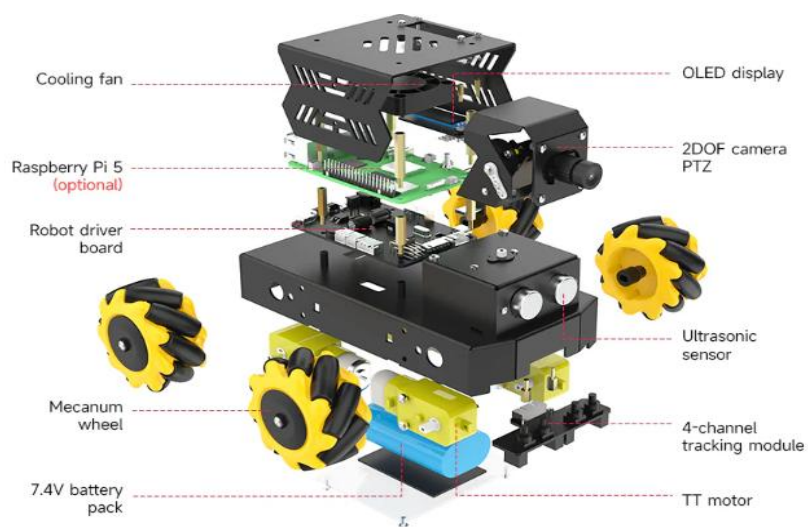
איור 4: הרכבת המכוננית, שלב 3



איור 5: הרכבת המכוננית, שלב 4



איור 6: מסך LCD



איור 7: רכב מלא ללא מסך



איור 8: התוצאה הסופית

3.2.2 תכנון אב

סביבת פיתוח :

סביבת הפיתוח שנבחרה לפרויקט זה מבוססת על שפת **Python** ועל מערכת ההפעלה **Linux** המותקנת על גבי בקר ה- **Raspberry Pi 5**. הבחירה ב-**Python** נובעת מהזמינות הרחבה של ספריות מתקדמות וחזקות בתחומי הרובוטיקה, הראייה הממוחשבת והבינה המלאכותית.

הפרויקט מתבסס באופן מרכזי על שתי ספריות קוד פתוח :

1. **OpenCV** : משמשת לביצוע משימות עיבוד תמונה בזמן אמת, כגון זיהוי צבעים ומעקב אחר אובייקטים.

2. **MediaPipe** : ספרייה של גוגל המאפשרת זיהוי מתקדם של תווי פנים, מחוות ידיים ותנועות גוף, ומספקת יכולות אינטראקציה טבעית עם המשתמש.

בנוסף, כלל המערכת מנוהלת תחת

ROS2 (Robot Operating System 2, Humble Edition), המספקת תשתית מודולרית לתקשורת בין רכיבי הרובוט (צמתים), ניהול נתונים, וכלים מתקדמים לוויזואליזציה ובקרה. סביבה זו נבחרה כדי לאפשר פיתוח הדרגתי, גמישות ויכולת הרחבה עתידית של הפרויקט.

בקרים:

הבקר הראשי שנבחר עבור הרכב הרובוטי הוא **Raspberry Pi 5**. בקר זה נבחר בזכות יכולות העיבוד החזקות שלו, המאפשרות לו להריץ אלגוריתמים מורכבים של ראייה ממוחשבת ובינה מלאכותית בזמן אמת. ה- **Raspberry Pi 5** משמש כיחידת העיבוד המרכזית (CPU) של המערכת, והוא אחראי על:

- איסוף ועיבוד נתונים מכלל החיישנים (מצלמה, חיישנים אולטרסוניים, חיישני עקיבה).
- הרצת סביבת **ROS2** והקוד שנכתב ב-**Python**.
- קבלת החלטות אוטונומיות המבוססות על הנתונים, כגון הימנעות ממכשולים, מעקב אחר אובייקטים ותגובה למחוות.
- שליטה על כלל הרכיבים ההיקפיים, כולל המנועים, מסך המגע, תאורת ה-**RGB** והזמזם, באמצעות לוח הרחבה ייעודי.

הבקר כולל יכולות תקשורת אלחוטית (**Wi-Fi**), המאפשרות שליטה מרחוק והעברת נתונים למערכות חיצוניות.

תכן חומרה :

- מצלמת USB 1MP + גימבל PTZ 2-DOF :** מצלמת HD המותקנת על מתקן ממונע המאפשר תנועה דו-צירית (אופקית ואנכית). המצלמה משמשת כ"עיניים" של הרובוט ומאפשרת יישומים כמו מעקב פנים, זיהוי מחוות וסריקת הסביבה. היא מתחברת לבקר באמצעות כבל USB ואינה דורשת התקנת דרייברים. (Plug-and-Play)
- **גלגלי Mecanum :** ארבעה גלגלים מיוחדים המאפשרים לרכב תנועה ב-360 מעלות. כל גלגל מורכב מגלילים קטנים המותקנים בזווית של 45 מעלות, מה שמאפשר תנועה קדימה, אחורה, הצידה, באלכסון וסיבוב במקום על ידי שליטה בכיוון ובמהירות של כל מנוע בנפרד.
- **מסך מגע קיבולי 7 אינץ' :** מסך ברזולוציה של 1024 X 600 המשמש כממשק משתמש אינטראקטיבי. הוא מאפשר להציג מידע חיוני, ויזואליזציה של זיהויים, ממשק גרפי (GUI) המדמה הבעות פנים, ולקבל קלט מהמשתמש ישירות על גבי הרובוט.
- **חיישן אולטרסוני :** חיישן למדידת מרחק המשמש למניעת התנגשות במכשולים. הוא פועל על ידי שליחת גל קול וחישוב המרחק על סמך הזמן שלוקח להד לחזור.
- **חיישן עקיבה 4 ערוצים :** מודול המכיל ארבעה חיישני אינפרא-אדום, המאפשר לרובוט לעקוב אחר קו שחור על משטח לבן (ולחיפך). החיישנים מזהים את החזר האור מהמשטח ובהתאם לכך מאפשרים ניווט מדויק לאורך מסלול מוגדר.
- **תצוגת OLED 0.91 אינץ' :** מסך קטן ברזולוציה של 128 X 32 אפיקסלים, המתחבר לבקר באמצעות פרוטוקול I2C. הוא משמש להצגת מידע מערכת חיוני ומהיר, כגון כתובת IP, עומס על המעבד, או מצב הסוללה.
- **לוח הרחבה (Expansion Board) :** לוח ייעודי המשמש כמתווך בין בקר ה-Raspberry Pi 5 לבין הרכיבים ההיקפיים השונים. הוא מספק חיבורים מסודרים למנועים, לסרורים, לחיישנים, לתאורת ה-RGB ולזמזום, ומנהל את אספקת המתח לרכיבים השונים.
- **סוללת ליתיום 2000mAh, V7.4 :** מספקת את האנרגיה הדרושה להפעלת המערכת באופן אלחוטי, ומאפשרת זמן עבודה רציף של כשעתיים-שלוש.

תכן תוכנה:

מודול AI לזיהוי ואינטראקציה חזותית: פותח מודול ב-Python המשתמש בספריות OpenCV ו-MediaPipe לעיבוד הווידאו המתקבל מהמצלמה בזמן אמת. הקוד אחראי על:

- **זיהוי פנים, מחוות ותנועות גוף:** האלגוריתם מנתח כל פריים מהווידאו כדי לזהות דפוסים מוכרים כמו פנים אנושיות או תנועות ידיים ספציפיות.

- **תגובה חזותית וקולית:** בעת זיהוי מוצלח, המערכת מפעילה תגובה, כגון שינוי צבע בתאורת ה-RGB, הצגת הודעה על המסך או השמעת צליל מהזמזום.

מערכת AI לשיחה אינטראקטיבית: כדי לאפשר אינטראקציה קולית טבעית, שולבה מערכת שיחה הבנויה משלושה רכיבים מרכזיים:

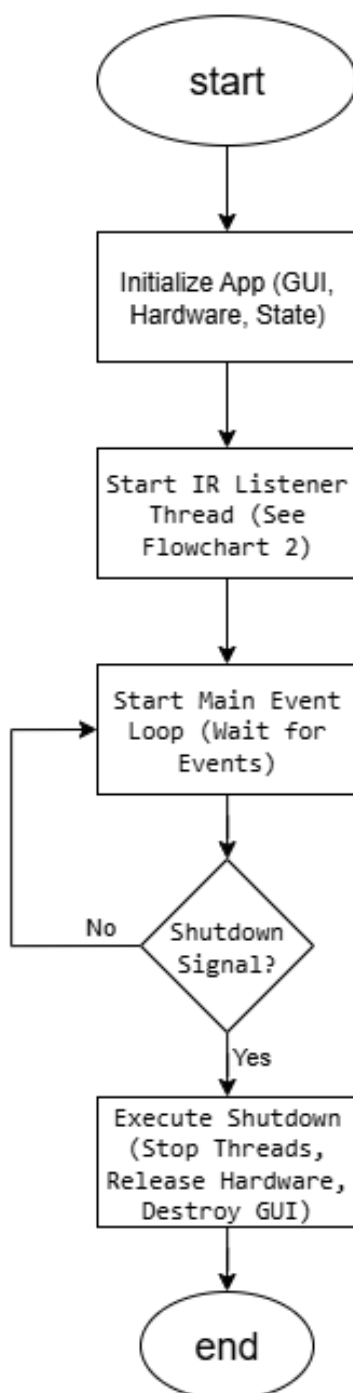
- **Vosk (דיבור לטקסט - STT):** ספריית קוד פתוח המשמשת להמרת הדיבור של המשתמש, הנקלט דרך מיקרופון, למחרוזת טקסט.
- **gemma2:2b (מודל שפה גדול - LLM):** מודל AI המקבל את הטקסט המומר, מנתח את כוונת המשתמש ומייצר תגובה טקסטואלית אינטליגנטית ורלוונטית.
- **piper-tts (טקסט לדיבור - TTS):** מנוע המקבל את התגובה הטקסטואלית ממודל השפה וממיר אותה לקובץ שמע קולי, המושמע דרך רמקולים.

ממשק תקשורת מבוסס ROS2: התקשורת בין חלקי התוכנה השונים (למשל, קוד עיבוד התמונה וקוד בקרת המנועים) ממומשת באמצעות ארכיטקטורת Publisher/Subscriber של ROS2. מודל זה מאפשר העברת נתונים (תמונות, פקודות, מידע מחיישנים) באופן אמין ויעיל בין "צמתיים" שונים במערכת. התקשורת מתבצעת על גבי רשת Wi-Fi בפרוטוקול TCP/IP.

מערכת בקרה ותנועה: נכתב קוד ייעודי השולט במנועי ה-DC של גלגלי ה-Mecanum. הקוד מקבל פקודות תנועה ומתרגם אותן למהירויות וסיבובים הנדרשים מכל אחד מארבעת המנועים. המערכת משלבת את נתוני החיישן האולטרסוני כדי לממש יכולות אוטונומיות של הימנעות ממכשולים.

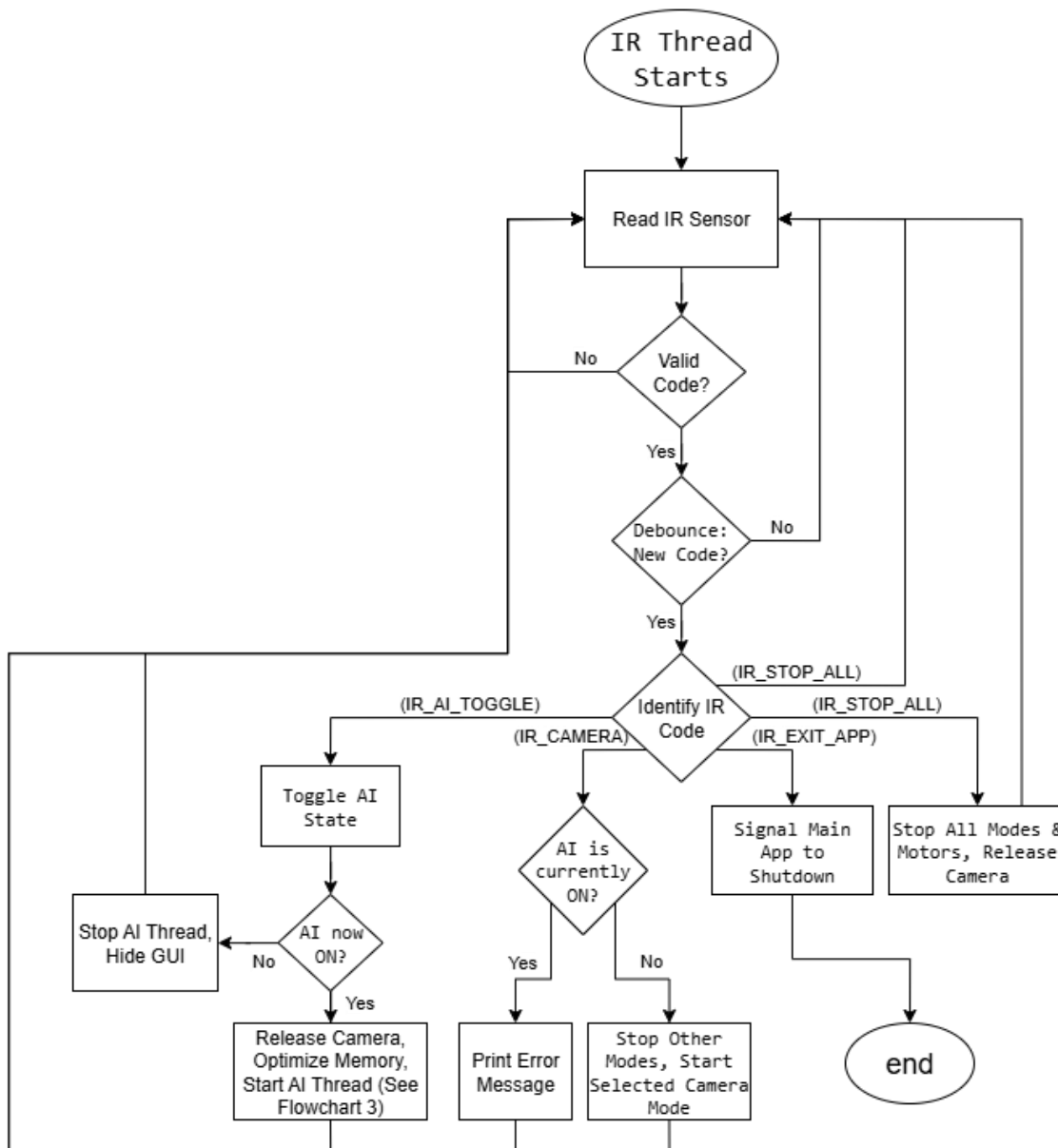
ממשק גרפי (GUI): פותח ממשק משתמש עבור מסך המגע בגודל 7 אינץ'. הממשק נועד להציג דמות אינטראקטיבית שמגיבה למתרחש בסביבה ומדמה רגשות. מערכת השיחה הקולית מסונכרנת עם הממשק הגרפי, כך שתנועות השפתיים והבעות הפנים של הדמות תואמות לדיבור המופק, ובכך נוצרת חווית שימוש טבעית וסוחפת יותר.

Main Application flow



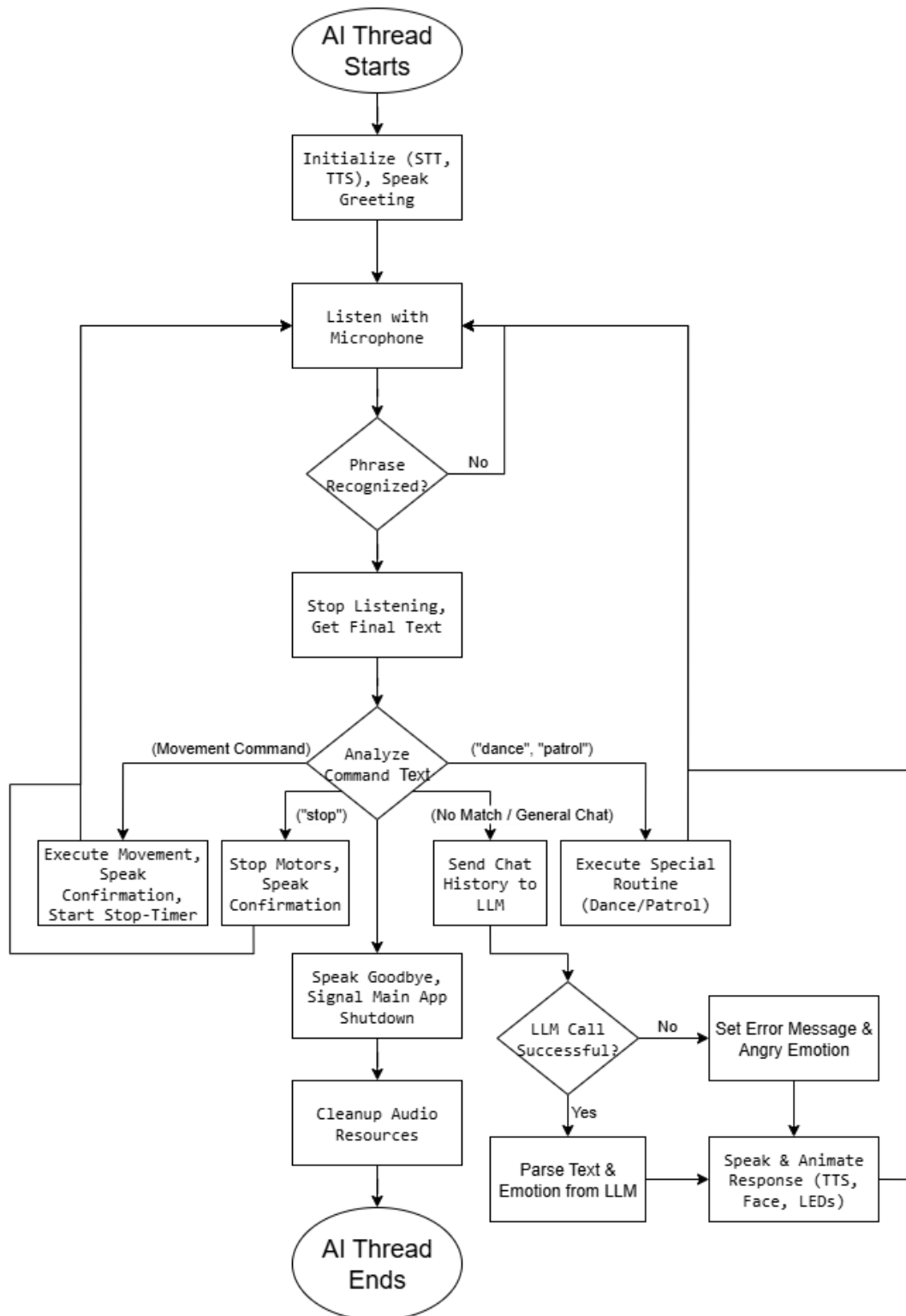
איור 9: תרשים זרימה כללי

IR Remote Trigger



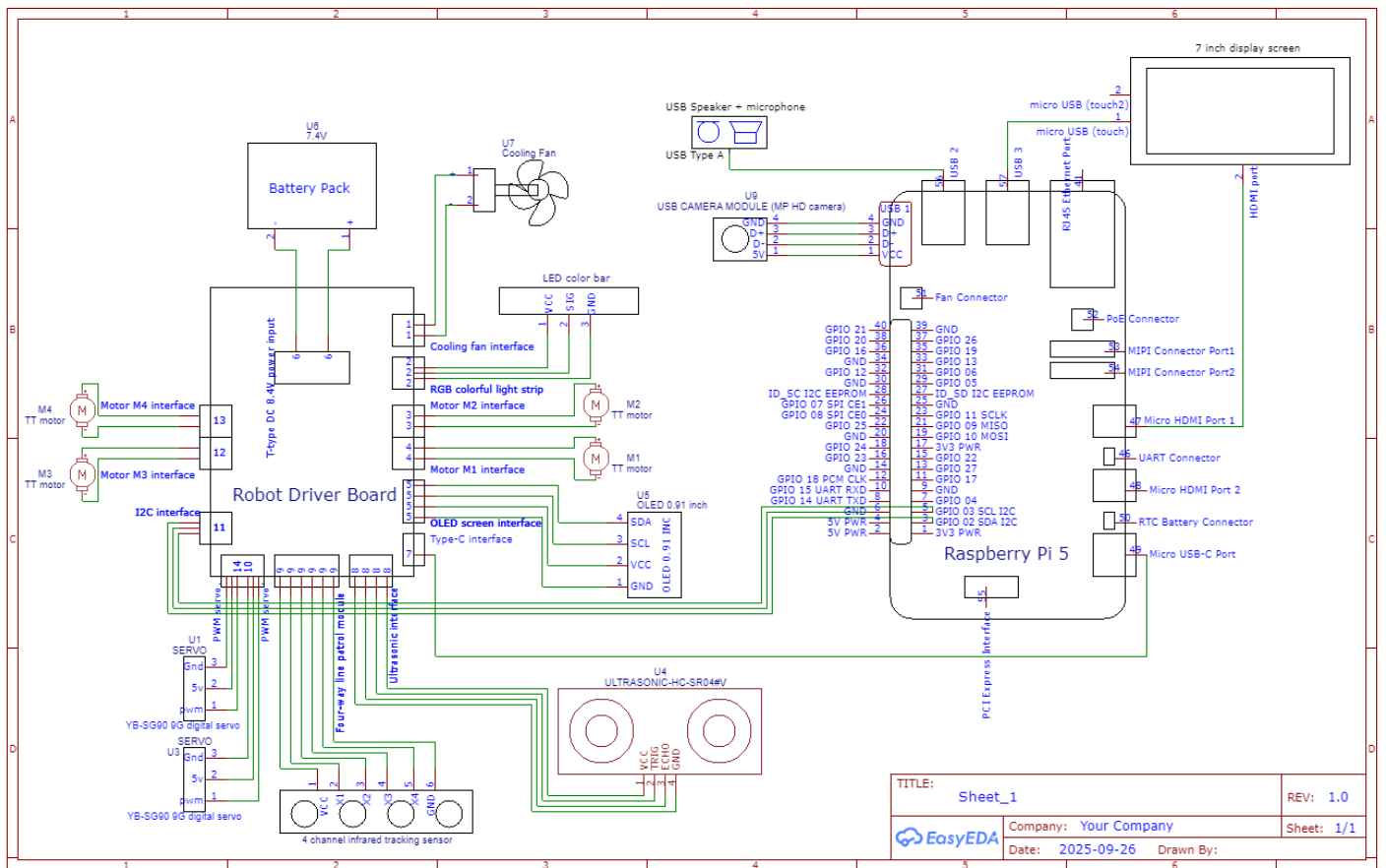
איור 10: תרשים זרימה IR

AI Chatbot Logic



איור 11: תרשים זרימה AI

3.2.4 סכמה חשמלית



איור 12: סכמה חשמלית

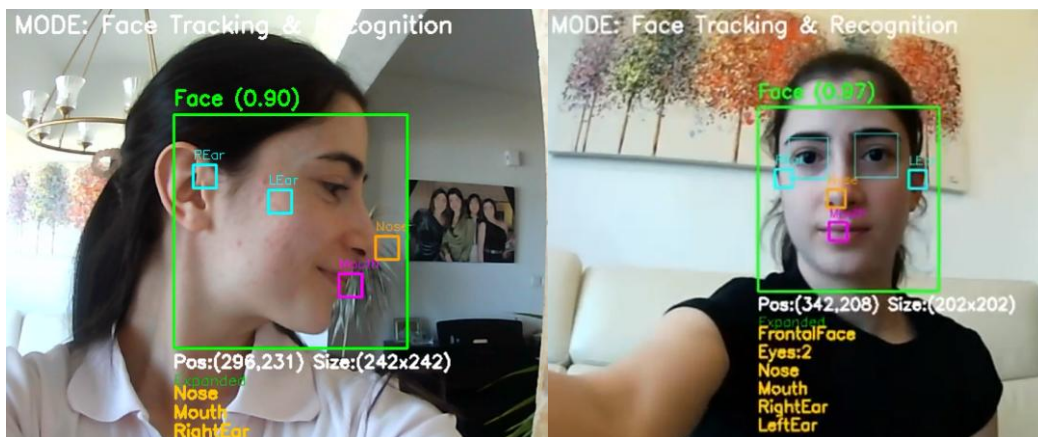
3.2.5 סימולציות

בחלק זה נבצע מספר בדיקות וסימולציות על היבטים ופונקציות שונות של "Marich". הבדיקה תחולק לשני חלקים עיקריים, הראשון יחקור את כל פונקציות הבינה המלאכותית הוויזואלית של המצלמה, והשני יעבור על פונקציות הבינה המלאכותית של LLM:

המלאכותית הוויזואלית:

- זיהוי פנים ומעקב:

כאן בדקנו את יכולתה של המצלמה לזהות את פנינו ולעקוב אחריהם. תחילה בדקנו אותה בתנאי תאורה טובים, הנה כמה תמונות מהבדיקה:



איור 13: זיהוי פנים באור חזק

הבדיקה הראתה לנו בבירור שזיהוי הפנים והמעקב עובדים מצוין, היא אפילו ניבאה היכן אמורים להיות חלקי פנים נסתרים.

לאחר מכן בדקנו את הפונקציה הזו בתנאי תאורה שונים:

רמת אור בינונית:



איור 14: זיהוי פנים באור בינוני

רמת אור נמוכה :

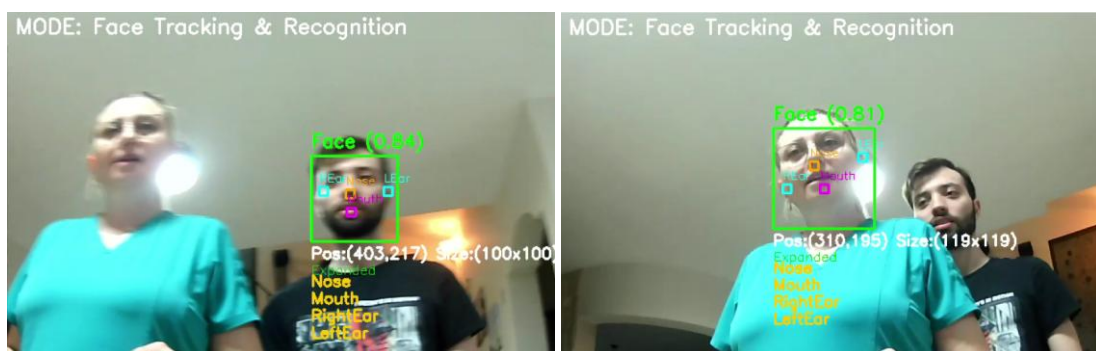


איור 15: זיהוי פנים באור נמוך

כפי שניתן לראות לעיל, בתנאי תאורה בינוניים, זיהוי הפנים והמעקב עדיין אמינים למדי, אם כי ניתן לראות דרך המספרים בתמונות שהתוכנית פחות בטוחה בהחלטתה.

ואז, כשאנחנו עוברים לתנאים חשוכים עוד יותר, כפי שניתן לראות בקבוצת התמונות האחרונה, התוכנה עדיין יכולה לזהות מדי פעם פרצופים, אבל היא גם מתחילה "לזהות" אותם גם במקומות בהם הם לא קיימים. בתנאי תאורה כאלה יכולות המעקב אינן אמינות.

הדבר האחרון שרצינו לבדוק בנוגע למעקב פנים הוא התנהגות התוכנית כאשר מספר פנים נמצאות בטווח הראייה שלה :

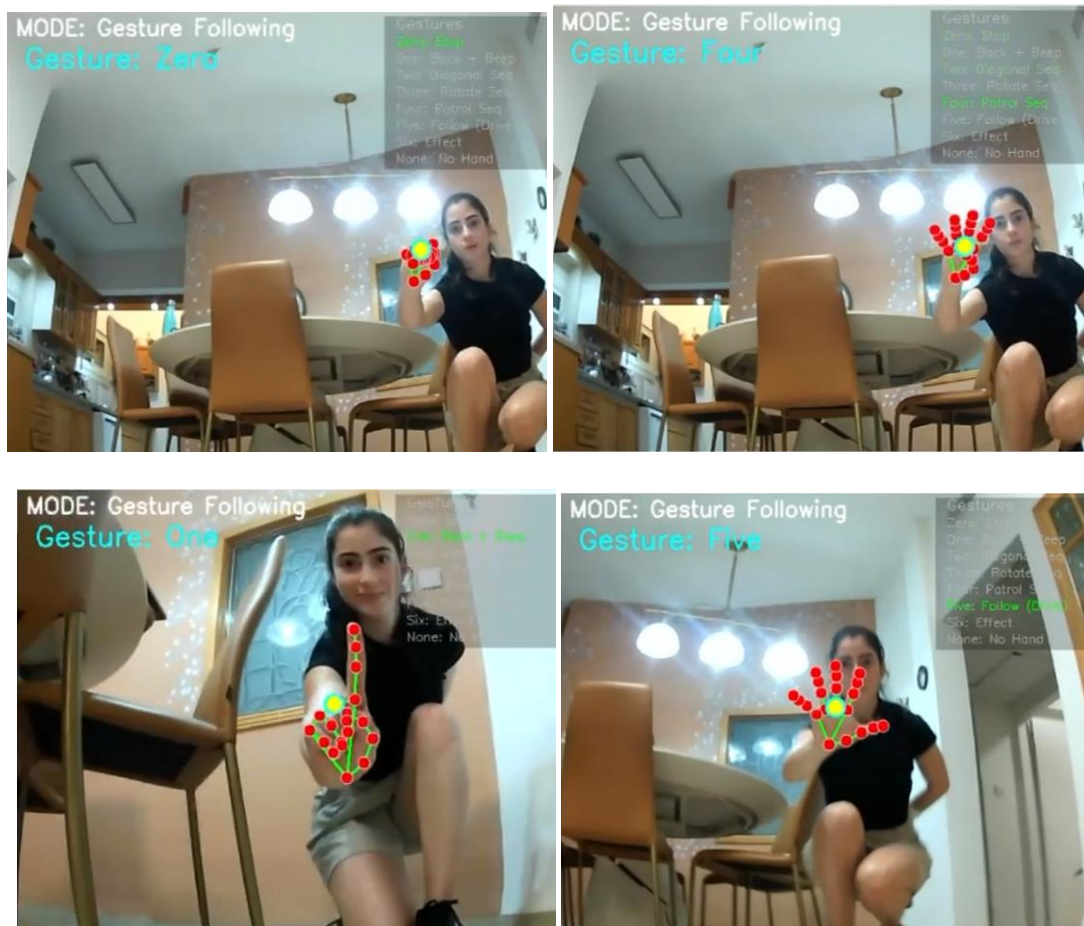


איור 16: זיהוי פנים כאשר מספר פרצופים זמינים

מהבדיקה ראינו שהתוכנית מזהה ומתמקדת בפנים אחד בכל פעם, תוך מעבר בין אנשים שונים מדי פעם. נראה כי נוכחותם של מספר אנשים לא הפריעה ליכולות המעקב שלה.

• תנועת יד עוקבת :

מצב זה עוקב אחר מיקום היד ומזהה כמה אצבעות מורמות. יש לו 6 מצבי התנהגות, בהתאם למספר האצבעות המורמות, מ-0 עד 6 (6 הוא כאשר רק האגודל והזרת מורמות). הנה כמה תמונות של התוצאות.

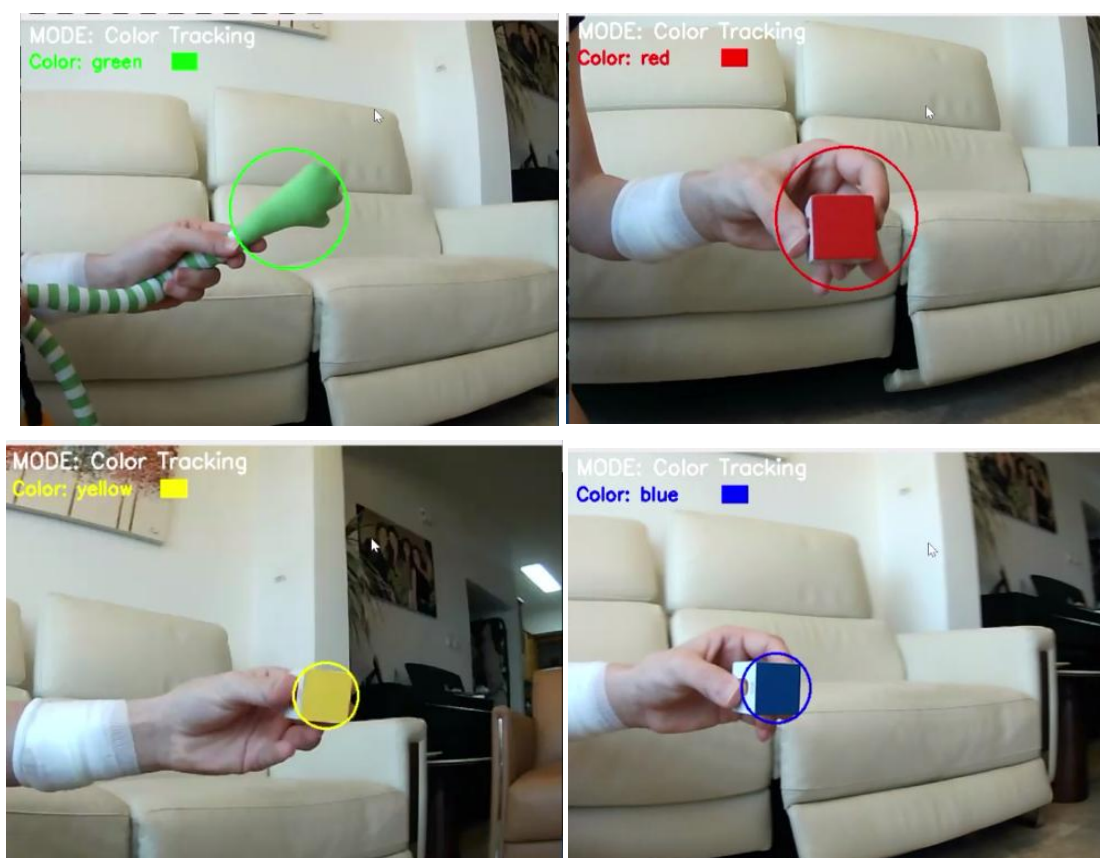


איור 17: תנועת יד עוקבת

המערכת הזו עבדה מצוין, אפילו ממרחקים גדולים יותר. הגיבה למחוות בצורה הנכונה, ולא איבדה הרבה קשר עם היד.

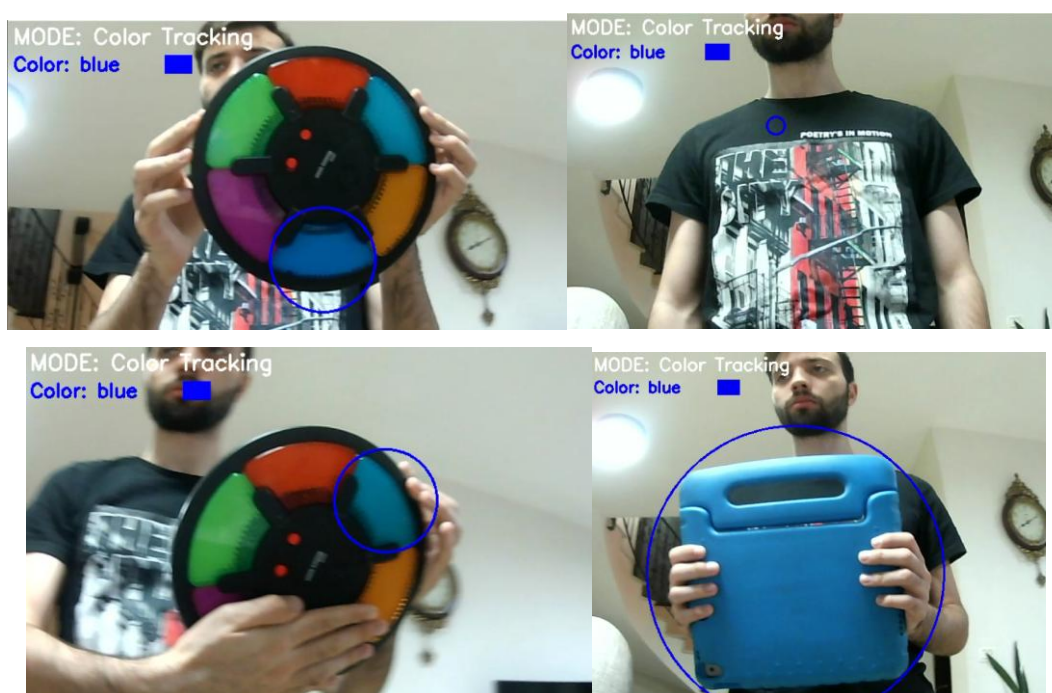
• מעקב צבעים :

כאן נוכל להגדיר איזה צבע נרצה לעקוב, והמצלמה שלך תעקוב אחריו, הנה כמה דוגמאות :



איור 18: זיהוי צבע

כאשר ניתנה לה אדום או כחול בסיסי, המערכת לא התקשתה לזהות אותו ולעקוב אחר כך. כדי לבדוק זאת לעומק בדקנו גוונים שונים של כחול:



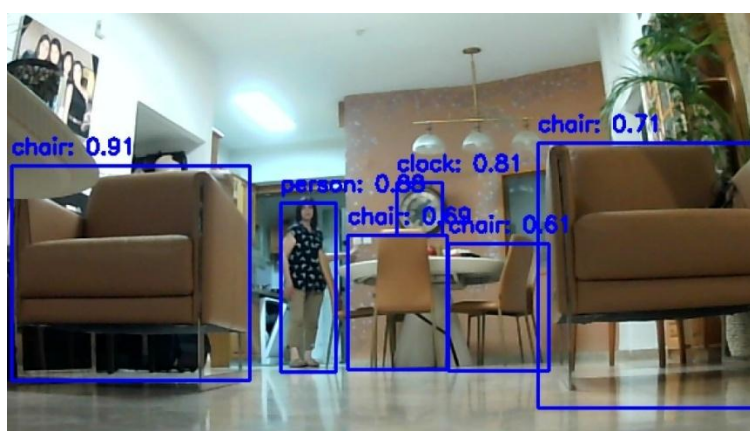


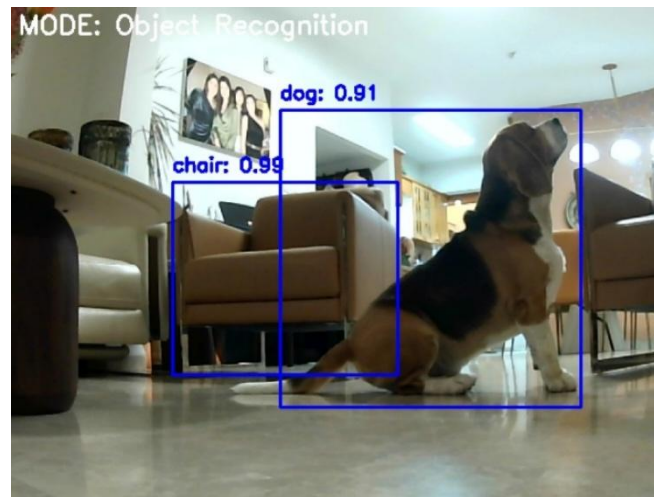
איור 19: זיהוי גוונים שונים של כחול

כפי שניתן לראות, גווני כחול במגוון רחב של גוונים זוהו בצורה נכונה (הכחולים השונים נראים שונים יותר זה מזה במציאות), זה מראה שהמערכת לא צפויה לפספס משהו שהוא כחול, אולם היא גם זיהתה את החולצה שלי ככחולה מספר פעמים במהלך הבדיקה, כשלמעשה היא שחורה, כך שאולי טווח הערכים הנחשבים ככחולים רחב מדי.

- זיהוי אובייקטים:

מצב זה יכול לזהות עצמים שונים בתצוגת המצלמה, כגון אנשים, כלבים, כיסאות, שעונים ועוד...





איור 20: זיהוי אובייקטים

במבחן שלנו, ראינו שיכולת הזיהוי, למרות היותה טובה למדי, יכולה לפעמים להתעלם מחפצים, כמו אורות או ציורים, או לזהות אותם בטעות. לדוגמה, המאמן זוהה כיושב ראש במבחן אחד, וכמאמן באחר. זיהוי שגוי זה אינו כזה גרוע.

- זיהוי לוחיות רישוי:

ולבסוף, מצב זה יכול לזהות לוחיות רישוי ואת המספרים הכתובים עליהן.

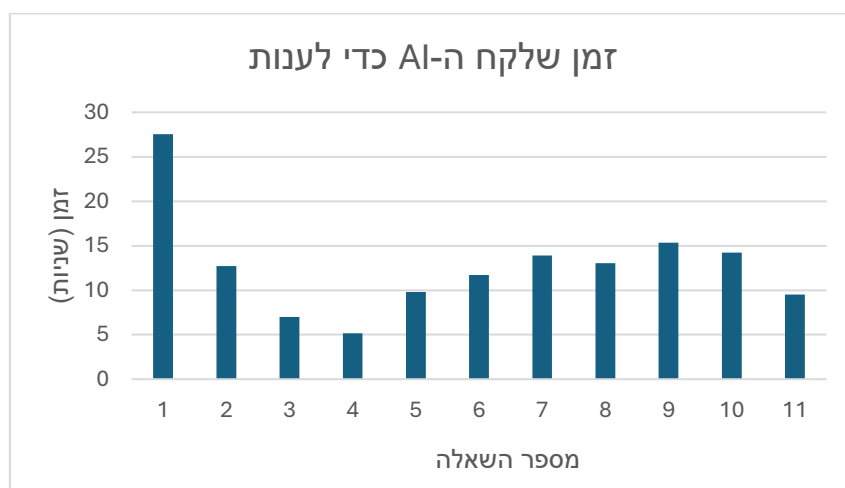


איור 21: זיהוי לוחות רישוי

כפי שאנו רואים, הוא מזהה את המספרים בצורה נכונה, אך לוחית הרישוי לא צריכה לנוע מהר וקרוב מספיק כדי שיזוהו.

הבינה המלאכותית של LLM:

- זמן התגובה של ה-LLM במבחן הראשון שלנו, טענו את מודל ה-LLM ושאלנו אותו 10 שאלות, בכל פעם תיעדנו כמה זמן לקח לו להגיב:



גרף 1: זמני התגובה של ה-AI

העמודה הראשונה מציינת את הזמן שלקח למודל LLM להיטען, ולכן זה לקח את הזמן הארוך ביותר. שאר העמודות מציינות את הזמן שלקח לשאלות שונות לקבל תשובה. ניתן לראות שהזמן נע בין מינימום של 5.15 שניות למקסימום של 15.37 שניות.

- רמת ההבנה: באמצעות אותן 10 שאלות לעיל, אנו מסווגים את השאלות לשאלות מורכבות ושאלות פשוטות. שאלה מורכבת היא שאלה שמסתמכת על תשובות קודמות (זיכרון של הבינה המלאכותית) והכילה סרקזם או בדיחות.

מורכבות השאלה	מס השאלות	ענה נכון	אחוז הדיוק
פשוט	5	5	100%
מסובך	5	3	60%
כל השאלות	10	8	80%

טבלה 2: דיוק התשובות

מהתוצאות ניתן לראות שכאשר נשאלת שאלה פשוטה, הבינה המלאכותית נותנת תשובה סבירה, אך כאשר נשאלת שאלה מורכבת יותר, הבינה המלאכותית מתחילה להתקשות ועלולה לעשות טעויות.

שמנו לב גם שההבדל בין השאלות המורכבות שקיבלו תשובה מדויקת, לבין השאלות המורכבות שקיבלו תשובה לא מדויקת, הוא שבסולם היו פחות מילים בניסוח השאלה, מה שנתן לבינה מלאכותית פחות הקשר לעבוד איתו, בעוד שהסולם הפורמלי היה ארוך יותר, מה שנתן לבינה מלאכותית יותר הקשר לעבוד איתו.

דיוק הרגש שהוצג היה קשור גם לדיוק התשובה, דבר זה מראה ששיטת הבחירה וההצגה של רגש מתאימה היטב ועובדת כמתוכנן.

לבסוף, מבחינת זיכרון, הבינה המלאכותית נועדה לזכור עד 3 שאלות ותשובות אחרונות, ובאמצעות הבדיקות שלנו ראינו שזה המצב.

3.3 בעיות ואתגרים הנדסיים

במהלך הפיתוח של הרכב הרובוטי החכם, התמודדנו עם מספר אתגרים הנדסיים משמעותיים שנבעו מהשילוב המורכב של חומרה מתקדמת, מערכות תוכנה מרובות ודרישות גבוהות לאינטראקציה בזמן אמת.

1. עיבוד בינה מלאכותית בזמן אמת על חומרה מוגבלת:

אחת ממטרות הליבה של הפרויקט היא אינטראקציה טבעית ומהירה. הפעלת מספר מודלי AI במקביל על בקר Raspberry Pi 5 הציבה אתגר חישובי משמעותי. המערכת נדרשה לעבד וידאו מהמצלמה בזמן אמת באמצעות MediaPipe (לזיהוי פנים, מחוות ותנועות גוף), ובמקביל להריץ את כל שרשרת האינטראקציה הקולית: זיהוי דיבור (STT), עיבוד שפה טבעית באמצעות מודל שפה גדול (LLM), והמרת טקסט לדיבור (TTS). האתגר היה לבצע אופטימיזציה לקוד, לנהל את משאבי המעבד והזיכרון ביעילות, ולבחור במודלים קלי משקל מספיק (כמו gemma2: 2b) כדי להשיג זמן תגובה מהיר שייצור חווית משתמש חלקה ולא מקוטעת.

2. כיול ודיוק של מערכת הנעה רב-כיוונית (גלגלי Mecanum):

גלגלי המקאנום מאפשרים לרובוט תמרון יוצא דופן של 360 מעלות, אך דיוק התנועה שלהם תלוי באופן קריטי בכיול ובקרה מדויקים. האתגר המכני והתוכני היה להבטיח שכל ארבעת המנועים פועלים במהירות מסונכרנת לחלוטין כדי לאפשר תנועה ישרה (קדימה או הצידה) ללא סחיפה (drift) או סיבוב לא רצוי. כל הבדל זעיר בביצועי המנועים, בחיכוך הגלגלים עם המשטח או בחלוקת המשקל של הרובוט היה צריך להיות מפצה על ידי אלגוריתם הבקרה. הדבר דרש תהליך ממושך של ניסוי וטעייה לכיוון פרמטרים בתוכנה כדי להגיע לתנועה חלקה, צפויה ואמינה.

3. אמינות מערכת הראייה הממוחשבת בתנאי סביבה משתנים:

ביצועי אלגוריתמי הראייה הממוחשבת (OpenCV ו-MediaPipe) תלויים מאוד בתנאי הסביבה. האתגר היה להבטיח זיהוי אמין של פנים, רגשות ומחוות בתנאי תאורה משתנים – אור חזק, צללים, או תאורה חלשה. רקעים מורכבים או הסתרה חלקית של פני המשתמש או ידיו היוו גם

הם מכשול. הפתרון דרש התאמות באלגוריתמים, כגון נורמליזציה של התמונה, ושימוש מושכל ביכולות הגימבל (PTZ) של המצלמה כדי למקם את המשתמש במרכז הפריים. עם זאת, נותר אתגר להתמודד עם מצבי קיצון, והמערכת נדרשה להיות מתוכננת כך שתזהה מתי תנאי הסביבה אינם מאפשרים זיהוי אמין.

4. הבנת פקודות קוליות בהקשר הנכון וסנכרון התגובה :

אתגר מרכזי היה להפוך את האינטראקציה הקולית לאמינה ואינטואיטיבית. התהליך כלל מספר שלבים : קליטת הדיבור (STT), הבנת הכוונה באמצעות מודל השפה (LLM), והפעלת תגובה מתאימה. נתקלנו בקושי בזיהוי פקודות מורכבות שהיו משולבות בתוך משפט ארוך. בנוסף, המערכת התקשתה לעיתים להבחין בין מילים שנאמרו כפקודה לבין מילים שנאמרו בהקשר אחר. לדוגמה, אמירת המילה "ימינה" כאישור ("נכון") גרמה לעיתים לרובוט לפנות ימינה פיזית. הפתרון דרש כיוול של מודל השפה וקביעת מילות מפתח ברורות יותר. אתגר נוסף היה לסנכרן את תחילת אנימצית הדיבור על המסך עם תחילת השמעת הקול ממערכת ה-TTS, כדי למנוע מצב שבו הדמות "מדברת" לפני ששומעים אותה, מה שפוגע בטבעיות של האינטראקציה.

5. אינטגרציה וסינכרון של מערכות חומרה ותוכנה מורכבות :

הפרויקט כלל שילוב של רכיבים רבים ומגוונים : בקר Raspberry Pi 5, מסך מגע 7 אינץ' (הדורש חיבורי HDMI ו-USB), מצלמת USB על גימבל PTZ ממונע, לוח הרחבה ייעודי, חיישנים (אולטרסוני, מעקב), מנועים, סרוו, תצוגת OLED ועוד. האתגר ההנדסי המרכזי היה לא רק לחבר פיזית את כל הרכיבים, אלא להבטיח תקשורת יציבה ונטולת קונפליקטים ביניהם. ניהול כל הרכיבים תחת מערכת ההפעלה לרובוטים (ROS2) דרש פיתוח "צמתים" (Nodes) ייעודיים לכל רכיב, וסינכרון מדויק של העברת הנתונים ביניהם כדי למנוע עיכובים (latency) וכשלים במערכת.

4. סיכום ודיון :

4.1 הצורך בפרויקט :

בעידן שבו טכנולוגיה ויצירתיות נפגשות, פרויקט זה נולד מתוך רצון לחקור את הגבולות שבין אינטראקציה אנושית למכונה. הצורך המרכזי היה ליצור פלטפורמה שהיא לא רק כלי לימודי מתקדם לסטודנטים וחובבי רובוטיקה ובינה מלאכותית, אלא גם ישות אינטראקטיבית המעוררת סקרנות, משחק והנאה.

הפרויקט עונה על מספר צרכים מרכזיים :

- **צורך חינוכי :** הוא משמש כפלטפורמה נגישה ומתקדמת המאפשרת למידה מעשית של טכנולוגיות מורכבות כמו ראייה ממוחשבת, בינה מלאכותית, ומערכות רובוטיות מבוססות ROS2.

- **צורך בידורי:** הרובוט תוכנן להיות בן לווייה מהנה ומשעשע. היכולת שלו להגיב למחוות, לזהות רגשות ולהציג הבעות פנים משלו הופכת אותו ל"צעצוע חכם" ואינטראקטיבי.
 - **צורך תפקודי:** מעבר למשחק, הרובוט פועל כעוזר אישי חכם. באמצעות שילוב מודל שפה מתקדם, הוא יכול לענות על שאלות, לספק מידע ולנהל שיחה, ובכך להדגים את הפוטנציאל של בינה מלאכותית בחיי היומיום.
- שילוב ייחודי זה בין חינוך, בידור ותפקוד יוצר מערכת רב-תכליתית המדגימה כיצד טכנולוגיה יכולה להיות אינטואיטיבית, מרתקת ומועילה בו-זמנית.

4.2 סיכום ומסקנות:

לסיכום, פרויקט זה עסק בתכנון, פיתוח ואינטגרציה של רכב רובוטי חכם, המשלב בין חומרה מתקדמת לתוכנה מורכבת. ליבת הפרויקט הייתה יצירת מערכת אינטראקטיבית המסוגלת לתקשר עם הסביבה האנושית באופן טבעי ואינטואיטיבי, הן באמצעות ממשק חזותי והן באמצעות שיחה קולית.

הפרויקט היווה אתגר הנדסי מורכב, שדרש שילוב בין דיסציפלינות שונות: מכניקה (הרכבת השלדה וגלגלי המקאנום), אלקטרוניקה (חיבור החיישנים, המצלמה והבקרים), ותוכנה (פיתוח אלגוריתמים לעיבוד תמונה, ניהול מערכת הפעלה לרובוטים, ומימוש מודלי בינה מלאכותית לזיהוי חזותי ושיחה).

במסגרת הפרויקט נחשפנו לעומק לעולם ה-AI והרובוטיקה. רכשנו ניסיון רב ערך בפיתוח בסביבת Python, בעבודה עם ספריות קוד פתוח חזקות כמו OpenCV ו-MediaPipe, וביישום ארכיטקטורת ROS2 לניהול מערכת מודולרית. ההתמודדות עם אתגרים כמו עיבוד נתונים בזמן אמת על חומרה מוגבלת וכיול מדויק של מערכות מכניות הקנתה לנו יכולות חשיבה יצירתית ופתרון בעיות. מעבר לערך הלימודי, הפרויקט סיפק לנו סיפוק אדיר ביצירת מערכת שקמה לתחייה ומגיבה לעולם באופן כה אנושי.

4.3 תוספות עתידיות:

הפלטפורמה שפותחה מהווה בסיס איתן להרחבות ושדרוגים עתידיים. להלן מספר כיוונים אפשריים להמשך פיתוח:

- **שיפור האינטראקציה והבינה המלאכותית:**
 - **הבנה קונטקסטואלית של פקודות:** שכלול מערכת זיהוי הדיבור כך שתדע להבחין בין שיחה רגילה לבין פקודה המיועדת לרובוט, על סמך הקשר המשפט ומילות מפתח.
 - **שילוב בין ראייה ממוחשבת למודל השפה (LLM):** יצירת אינטגרציה עמוקה יותר, כך שתגובותיו של הרובוט יושפעו לא רק מהשאלה הקולית, אלא גם

מגורמים בסביבה הנקלטים דרך המצלמה, כמו הבעת הפנים של המשתמש או זיהוי אובייקטים בחדר.

- **זיהוי וזיכרון אישי:** הוספת יכולת לזהות משתמשים ספציפיים באמצעות זיהוי פנים, ולפתח "זיכרון" ארוך טווח המאפשר לרובוט לזכור אינטראקציות קודמות ולהתאים את תגובותיו באופן אישי.

• **שיפור ביצועים וחומרה:**

- **האצת תגובות ה-AI:** בחינת שימוש בחומרה ייעודית להאצת חישובי בינה מלאכותית (כמו Google Coral או Nvidia Jetson) כדי לקצר את זמני התגובה של המערכת, במיוחד במודל השפה.

• **הרחבת הפונקציונליות:**

- **הוספת משחקים ורגשות:** פיתוח יישומים נוספים כמו משחקים אינטראקטיביים המשלבים תנועה וזיהוי, והרחבת מגוון ה"רגשות" והתגובות שהרובוט יכול להביע.
- **הרחבת מאגר הפקודות:** הוספת פקודות קוליות ומחוות חדשות שיאפשרו שליטה רחבה יותר על תנועות הרובוט ופעולותיו.

4.4 עמידה בדרישות:

דרישות	בוצע בפרויקט
כתיבת הצעת הפרויקט.	בוצע
תכנון, הרכבה וחיבור כל רכיבי החומרה (שלדה, גלגלים, מצלמה, חיישנים ומסך).	בוצע
מימוש יכולות ראייה ממוחשבת באמצעות	בוצע
מימוש מערכת ניווט ובקרה (שליטה בגלגלי	בוצע
פיתוח ממשק משתמש גרפי)	בוצע
בניית משחקים אינטראקטיביים לשיפור האינטראקציה עם המשתמש.	בוצע
שילוב יכולות למידת מכונה)	בוצע
אינטגרציה מלאה בין כל רכיבי החומרה והתוכנה במערכת.	בוצע
בדיקת המערכת בתנאים שונים, איתור תקלות וביצוע אופטימיזציה.	בוצע
כתיבת ספר פרויקט מסכם.	בוצע

טבלה 3: עמידה בדרישות

5. ביבליוגרפיה:

1. DeepMind. (2024). *Gemma 2: Open language models*. Retrieved from <https://deepmind.google/discover/blog/gemma-2-open-models/>
2. MediaPipe. (2024). *MediaPipe Solutions*. Google AI. Retrieved from <https://developers.google.com/mediapipe>
3. OpenCV. (2024). *OpenCV-Python Tutorials*. OpenCV.org. Retrieved from <https://docs.opencv.org>
4. Piper TTS. (2024). *piper-tts: A fast, local neural text to speech system*. GitHub repository. Retrieved from <https://github.com/rhasspy/piper>
5. Raspberry Pi Foundation. (2024). *Getting started with Raspberry Pi 5 and Raspberry Pi OS*. Retrieved from <https://www.raspberrypi.com/documentation/>
6. ROS 2. (2023). *ROS 2 Documentation: Humble Hawksbill*. Open Robotics. Retrieved from <https://docs.ros.org>
7. Vosk. (2024). *Vosk Speech Recognition Toolkit*. Alpha Cephei. Retrieved from <https://alphacephei.com/vosk>
8. HC-SR04 Ultrasonic Sensor. (n.d.). *Technical datasheet*. Retrieved from <https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf>
9. Mecanum Wheel Kinematics. (2020). *Mecanum wheel kinematics and dynamics – tutorial resources*. Retrieved from <https://automaticaddison.com/mecanum-wheels-kinematics/>
10. PyImageSearch. (2024). *OpenCV + Raspberry Pi tutorials*. Retrieved from <https://pyimagesearch.com/category/raspberry-pi/>

6. נספחים:

בקובץ zip מצורפים הקבצים

בנוסף ב google drive יש תמונות וסרטונים :

https://drive.google.com/drive/folders/12vms8W5YqA-3uK9dnWR5Rz2Ld92eqmh?usp=drive_link