בס"ד איילה אילוז

**הצעת פרוייקט/ מכונית אוטונומית:**

***רקע תאורטי:***

רכב אוטונומי זהו רכב שלרוב פועל בכוחות עצמו ואינו נזקק לעזרתו של הנהג.

המכונית האוטונומית משתמשת בחיישנים שונים שקולטים מידע על הסביבה, מעבדים אותו

באמצעות אלגוריתמים מתקדמים של בינה מלאכותית כגון yolo ומנווטים את המכונית בהתאם

תוך שמירה על בטיחותו של הנהג ובטיחות הסובבים אותו.

חיישנים אלה כוללים: מד מהירות על הגלגל, GPS, מצלמות, לידאר ורדאר.

כ-23 מיליון בני אדם עובדים בתעשיית הרכב העולמית, ורק כרבע מהם מועסקים ביצרניות הרכב

עצמן. השאר הם עובדי ספקיות המשנה שמייצרות רכיבים שונים לתעשייה – בלמים, צמיגים,

מערכות מחשוב וכדומה.

לאלה מצטרפות בשנים האחרונות חברות הייטק, על רקע פיתוח המכוניות האוטונומיות.

שרשרת האספקה של יצרניות המכוניות מורכבת ממאות חברות.

חברות השלב הראשון (טיר 1) מוכרות את המוצרים שלהן ישירות לחברות הרכב,

וחברות השלב השני (טיר 2) הן ספקות רכיבים לשלב הראשון וליצרניות הגדולות.

מטרתה של המצאת הרכב האוטונומי היא לשפר את הבטיחות בדרכים- כיוון שהמכונית

מתוכנתת עפ"י חוקי התנועה ומצייתת להם, לצמצם את הגודש בכבישים, להקל על פעילות

הנהג ולשמור על איכות הסביבה.

**יתרונות רכב אוטונומי:**

* **פינוי שטחי חניה רבים לטובת בניה-** לפי הערכות עד 2030 עשרה אחוזים מהרכבים

שמיוצרים בעולם יהיו אוטונומיים, וסביר להניח שעד 2035 מחצית מכלי הרכב כבר יהיו

אוטונומיים. סביר להניח שמספר כלי הרכב הנעים בכבישים יופחת ורבים יוותרו על

רכישת ואחזקת מכונית פרטית. השחרור מצורכי חניה יפנה מיליוני מטרים של נדל"ן.

מכונית אחת תוכל לשמש הרבה אנשים. וזאת בגלל שכאשר אדם צריך לנסוע, הוא מזמין מכונית,

והמכונית הכי קרובה אליו בסביבתו תגיע אליו. כך שלו היה לו רכב הוא היה חונה המון שעות

סתם ותופס שטח חניה לחינם.

* **מניעת תאונות וטעויות אנוש בנהיגה-** על פי נתוני משרד התחבורה האמריקאי 9 מתוך 10

תאונות מתרחשות בשל טעויות נהג, אבל המחקר קובע כי כשליש מהתאונות נובע מצירוף

נסיבות שגם תגובה מהירה ומדויקת לא הייתה יכולה למנוע את התרחשותן. למשל הולך רגל

שקופץ פתאום לכביש קרוב מדי מכדי שרכב יספיק לבלום. או מצבים אחרים שבהם אי-אפשר

היה לצפות את ההתנהגות של נהג אחר. בוחני המכון, שמבצעים גם מבחני ריסוק למכוניות

חדשות, בחנו 5,000 תאונות.

לדבריהם, רכב אוטונומי לא יסבול מהסחת דעת, עייפות או שכרות שמהם סובל נהג אנושי,

ותמיד יוכל להגיב במהירות למצבי חירום, כמו כן מצויית לחוקי התנועה, אבל תאונות רבות

מתרחשות בנסיבות שגם מכ"ם, מצלמה וחיישנים אחרים לא יכולים לצפות, או דווקא בשל

העובדה שלעיתים נדרשת חשיבה יצירתית ועקיפת החוקים דבר שקיים רק אצל בן אנוש.

* **שמירה על סביבה נקייה-**מכוניות אוטונומיות רבות מונעות על ידי מנועים חשמליים,

המייצרים אפס פליטות בנקודת השימוש. זה מפחית את זיהום האוויר ואת פליטת גזי

החממה, ועוזר להילחם בשינויי האקלים.

כמו כן, מכוניות אוטונומיות יכולות להשתמש באלגוריתמים מתקדמים כדי לייעל את המסלולים שלהן, להפחית עומס ופקקים ולשמור על מהירות נסיעה קבועה. זה עוזר למזער את צריכת הדלק והפליטה על ידי הימנעות מתנועת סרק מיותרת ותנועת עצירה וסע.

* **צמצום העומס בכבישים-** כל אדם שירצה לנסוע, יזמין אליו מכונית דרך האפליקציה והמכונית הפנויה בסביבתו תגיע אליו באופן אוטומטי. דבר זה יצמצם את כמות הרכבים כיוון שהרבה יעדיפו להשתמש במודל חדשני זה, ללא עלויות אחזקת הרכב.

**חסרונות רכב אוטונומי:**

* **הסקת מסקנות שגויה-**רכב אוטונומי מגיב מהר יותר מרכב המופעל אנושית, ועלול גם לנוע בפתאומיות. כמו כן, הוא מתוכנת לציית תמיד לחוקי התנועה והתמרורים, אך קיימים מצבים שבהם התנהלות חוקית 'לפי הספר' עלולה להוביל לתאונה.

כשמשתלבים בכביש מהיר, לדוגמה, בעוד שאר המכוניות נעות במהירות גבוהה יותר מהמותרת, מצפים שגם האוטונומית תתאים את מהירותה, כדי למנוע מנהג שנע לא מרוכז מאחור להתנגש בה.

בעולם הבינה המלאכותית אין ביכולת המתכננים לצפות מראש כיצד תגיב המערכת הממוחשבת ובהתאם אין באפשרותן לצפות כיצד יפעל הרכב בכל מצב משתנה.

* **הגברת העומס בכבישים-**עלות הנסיעה ברכב זה תהיה זולה כיוון שהמכונית נוסעת ללא נהג,

וכך בכל נסיעה אנו חוסכים תשלום לנהג, ואז רבים יעדיפו לנסוע במכונית האוטונומית ולא בתחבורה ציבורית.

דבר זה יגרום דווקא לעליית מספר כלי הרכב בכבישים ולא הפחתתם.

* **חשיפה לפרצות ומתקפות סייבר-** שליטה על מערכת הניווט, סנכרון מדיה, ביצוע שיחות טלפון, המזגן מתחיל לעבוד מעצמו, לוח המחוונים מתחיל להבהב ולייצר התרעות שונות ומשונות, חוסר יכולת לצאת מהרכב מכיוון שהדלתות ננעלות, שליטה על מערכות בקרת כלי הרכב, השבתת הבלמים והפסקת פעולת המנוע, הפעלת כריות אוויר תוך נסיעה מהירה, הקרנת תמונות המציגות דמויות על הכביש או על שלטי חוצות אשר אלגוריתמי ניתוח תמונה ממצלמות המכונית מזהים אותם כבני אנוש ובעקבות כך נעצרת נסיעת הרכב או פונה לנתיב אחר, השתלטות על רכב המוביל חומרים מסוכנים ופיצוצו במקום הומה אדם.

**מנגנונים חכמים:**

על הרכב האוטונומי מותקנים מגוון רחב של חיישנים העוזרים לחקות את הראיה האנושית, המוח האנושי, לקלוט את מפת הסביבה ולבצע חישובים רלוונטים כגון מרחק מעצמים, מהירות וכו'...

**:LIDAR** חיישני לייזר שמזהים אובייקטים שנמצאים לפני הרכב ומכוונים אותו. LIDAR הוא שילוב של המילים אור – LIGHT ומכ"ם –RADAR . בכלי רכב אוטונומיים משתמשים בLIDAR כדי למפות מכשולים בתוואי הנסיעה וכדי לנווט את הרכב בצורה בטוחה על הכביש. טכנולוגיית מדידת המרחק של ה- LIDAR מבוססת על הארת המטרה בקרן לייזר.

ה- LIDARמשגר בעצמו את האור הממוקד לכמה כיוונים בסביבת המכונית, הקרניים פוגעות בעצמים סביב המכונית וחלקן מוחזרות מהם אל החיישן לצורך חישוב מרחק העצם מהרכב. כמו כן הוא מחזיר ענן נקודות מהסביבה המנותח ע"י אלגוריתם לפילוח ענן נקודות וכך מזהים מכשולים בדרך, שפת מדרכה, גדרות ביטחון ורכבים אחרים. אין לו בעיה בחושך והוא טוב ממצלמה באור חזק, אבל הוא לא רואה צבעים. החיישן מסוגל למדוד את המרחק של הרכב מעצמים בסביבתו בדיוק של עד שני סנטימטר באמצעות הבזקי אור קצרים שמשכם חלקיקי שנייה.

חיישני LIDAR משלימים את אמצעי הצפייה והזיהוי ומאפשרים לרכב להישאר בנתיב ולבלום מייד כשמאותר מכשול. בנוסף, גלגלי המכונית מצוידים בחיישני אולטרה-סאונד שמזהים מכשולים גם בגובה הקרקע ומאפשרים לרכב לחנות בבטחה.

**מעבד האות:** מעבד האות הוא מרכיב קריטי בביצועי המכ"ם , ותפקידו לבצע עיבודים מתמטיים כמו חילוץ האות מתוך רעשי רקע, או להפריד את האנרגיה הנקלטת לפי תדרי הדופלר כדי לחלץ את האות החוקי מהמטרה.

**מצלמה**- מצלמה משוכללת היא חלק ממערך החיישנים ומסוגלת לזהות רמזורים ואת צבע האור ברמזור, תמרורים בסביבה, סימונים על הכביש, הולכי רגל, מכשולים ואף לחשב את המרחק אליהם. אולם היא רגישה לאור חזק ומתקשה גם בחשיכה. המצלמות, ובכללן ההיקפיות, משמשות לרוב לזיהוי אובייקטים במרחקים קצרים יחסית ובסיטואציות לא מורכבות במיוחד. כדי לחשב את המרחק היחסי מהמכונית שנוסעת לפני הרכב האוטונומי, מחשב המצלמה מסיק את המרחק באמצעות גיאומטריה. הוא לוקח בחשבון את אורך המוקד שבו היא מצולמת, את הגובה מפני הקרקע שהמצלמה מותקנת ואת המרחק היחסי של גלגלי הרכב המצולם. פרמטרים אלה מאפשרים לחשב את המרחק בין המצלמה למכונית שהיא מצלמת.

באמצעות אלגוריתם yolo-המנתח תמונה בזמן אמת ניתן לזהות את העצמים במרחב.

**מכ"ם-** מכ"ם אינו רגיש למזג אוויר אבל סובל מדיוק נמוך. הרדאר, המבוסס על גלי RF , מיועד לזהות אובייקטים בטווחים הרחוקים ובמזג אוויר קשה, בעיקר באזור המצוי לפני הרכב. יצרני הרכב בחרו להפעיל את המכ"מים בתדר 77 גיגה הרץ, ושואפים לשפר את הרזולוציה שיכולה ממש לייצר תמונה. זה קריטי כדי לזהות אופנוע בכביש מהיר, גם כשהוא מתמזג מבחינת החיישנים עם רכב אחר, וכדי לזהות הולכי רגל ואופניים בקרבת מכוניות. האתגר הכי קשה של הרדארים הוא לזהות הולכי רגל ורוכבי דו-גלגלי, גם על רקע כלי רכב אחרים. כמו כן, לזהות את הגובה של חפץ במרחק של 150 מטר, כדי לדעת אם זו פיסת קרטון שלא מפריעה לנו או ארגז שלא נרצה להתנגש בו.

**בעולם קיימים רכבים בעלי רמות שונות של אוטונומיה:**

* **רמה 0: ללא אוטומציה בנהיגה –** הנהג מחויב בכל פעולות הנהיגה, כפי שאנו מכירים זאת בעשורים האחרונים. המערכת האוטומטית מציגה אזהרות ועלולה להתערב לפרקי זמן קצרים, אך אין לה שליטה מתמדת על הרכב. אין לו אפילו בקרת שיוט, ובוודאי לא בקרת שיוט אדפטיבית.
* **רמה 1: סיוע לנהג** – קיימת ברכב מערכת סיוע נקודתית לנהג. כלי רכב ברמה 1 יצאו לשוק החל מסביבות שנת 2010. הנהג והמערכת האוטומטית חולקים שליטה ברכב. לדוגמה מערכת שבה הנהג שולט על ההיגוי, והמערכת האוטומטית שולטת על כוח המנוע כדי לשמור על מהירות מוגדרת (בקרת שיוט), כדי לשמור על המהירות ולשנותה (בקרת שיוט מסתגלת), סיוע בחניה, שמירה על נתיב הנסיעה ובלימת חירום למניעת התנגשויות. בישראל קיימת כבר מתאריך 1.1.2018 חובת התקנת מערכת בטיחות של בקרת סטייה מנתיב והתרעת שמירת מרחק.
* **רמה 2: אוטומציה חלקית** – מכונית שבה פעילות שתי מערכות או יותר של בטיחות ואוטונומיה, אשר מסוגלות, למשל, לטפל בו זמנית בהיגוי בתאוצה ובבלמים. יצאו לשוק החל משנת 2016, וברמה זו נדרש עדיין הנהג לבחון את תנאי הדרך ולהגיב בהתאם. על הנהג לפקח על הנהיגה ולהיות מוכן להתערב בכל עת אם המערכת האוטומטית. אפשר לעקוב אחר עיניו של הנהג באמצעות מצלמות כדי לאשר שתשומת ליבו מופנית לנסיעה. למערכות החיישנים החכמים ולמצלמות ההיקפיות ברמה זו של אוטומציה חלקית מתווספות מערכות נרחבות יותר, שנכללים בהן חיישנים המקיפים את השטח המת בשדה הראייה, מערכות התרעה על עקיפה מימין ואפילו אמצעי חניה אוטומטיים. לעיתים קרובות הרכב מצויד בבקרת שיוט ובשליטה חלקית על ההיגוי והבלימה.
* **רמה 3: אוטומציה מותנת** – המערכות של המכונית ברמה זו בשלות מספיק כדי שהנהג לא יידרש לפקח על תנאי הדרך "עיניים סגורות", אולם נדרש להיות זמין לנהיגה במקרה חירום. ברמה זו מערכות משולבות נהג-מחשב. הרכב יטפל במצבים המצריכים תגובה מיידית כגון בלימת חירום. שווק לראשונה ע"י חברת אודי באוקטובר 2018. בספר הרכב של הדגמים כתוב במפורש שהתכונות המופעלות בדגם דורשות פיקוח אקטיבי של הנהג ולא הופכת את הרכב לאוטונומי. אפשר לחשוב על המערכת האוטומטית כעל נהג שותף שמתריע באופן מסודר לנהג שתורו לנהוג. הנהג יכול להפנות את תשומת ליבו בבטחה ממשימות הנהיגה. הוא יכול לשלוח הודעות טקסט או לצפות בסרט. המערכת עומדת בתקנות הבינלאומיות של מערכת לשמירת נתיבים אוטומטית –. ALKS מערכות השליטה ברכב עוברות ממחשב הרכב אל הנהג על פי המקרה, מפגעים בכביש ותנאי מזג אוויר. לדוגמא, בתנאים שאינם מצריכים תשומת לב מיוחדת, כגון כביש מהיר, יוכל המחשב להיכנס לפעולה, עם זאת לעיתים קרובות עדיין נדרשת התערבות אנושית.
* **רמה 4: אוטומציה ברמה גבוהה** – יכולת נהיגה אוטונומית מלאה בתנאים מסוימים או בנסיבות מיוחדות. יש צורך בנהג זמין למצבי חירום ואת תשומת ליבו לבטיחות. נהיגה עצמית של הרכב נתמכת רק באזורים מרחביים מוגבלים. מחוץ להם על הרכב להיות מסוגל לתפעל את הנסיעה ואת החניה בבטחה. רמה 4 מתאימה לדוגמא למונית רובוטית או שירות משלוחים רובוטי הפועלים במיקומים מוגדרים. מאחר וברמה זו הנהג יכול לשלוט על מערכות הרכב, נשאלת השאלה המשפטית – מתי המכונית הורתה לו להתערב, עד כמה בכלל יש לו שיקול דעת והאם מלכתחילה ההתערבות הייתה נחוצה. ברמה זו מחשב הרכב יכול לפקד על כל משימות הנהיגה. הוא יעבור למצב זה בשטחים ללא תנועה כבדה של רכבים והולכי רגל או בכבישים מהירים. בכל שלב הנהג יכול לשלוט בכל משימות הנהיגה.
* **רמה 5: אוטומציה מלאה** – הרכב יהיה מסוגל להחליף את הנהג בכל תנאי הדרך, ואין צורך במעורבות נהג למצבי חירום. מדובר באמת בטייס אוטומטי שמסוגל לשלוט לבדו במכונית. דוגמה לכך היא מונית רובוטית שעובדת בכל תנאי מזג האוויר ובכל הדרכים ללא מגבלות. הנהג לא חייב לשבת ברכב. התערבותו אינה נדרשת כלל, ולכן אין צורך בהגה או בדוושות. אפשר להגדיר תרחישים קבועים ולשלוח את המכונית עם הילדים לבית הספר, להזמין את הרכב לאיסוף ממקום מסוים ועוד משימות. השליטה ברכב נעשית באמצעות טכנולוגיות LOT שמאפשרות להזמין אותו לכל נקודה דרך אפליקציה ייעודית.

כיום ניתן לתכנת רכב אוטונומי בעל אוטומציה ברמה גבוהה (רמה 4).

וזאת כיוון שעם כל הטכנולוגיה המתקדמת והמשוכללת, עדיין יש מחסור במכשירים משוכללים מספיק שיוכלו לעמוד בכל התנאים הנדרשים.

לדוגמא:

* **מצלמה-**
* עדשות מצלמות שאינן נקיות מעבירות אינן באמינות את המידע הוויזואלי.
* למצלמות יש מגבלות הדומות לאלה של העין האנושית, כלומר הן עשויות "לראות" גרוע יותר למשל בשלג או גשם עזים, ערפל סמיך, סופות אבק כבדות ופתיתי שלג. בתנאים כאלה, התפקודים של מערכות תלויות מצלמה עלולים להיות מופחתים במידה משמעותית או להתנתק באופן זמני. אור מתקרב בעצמה גבוהה, השתקפויות בכביש המהיר, שלג או קרח על פני הכביש, משטחי כביש מלוכלכים או סימוני נתיבים לא ברורים יכולים גם הם להפחית במידה משמעותית את תפקוד המצלמה כשמשתמשים בה לסריקת הכביש לזיהוי הולכי רגל, רוכבי אופניים, בעלי חיים גדולים וכלי רכב אחרים.
* **לידאר-**
* ביצועי LiDAR יכולים להיות מושפעים מתנאי מזג האוויר, כגון גשם, ערפל ושלג.

זה יכול להגביל את השימוש בו ביישומים חיצוניים באקלים מסוימים.

* **רדאר**-
* היכולת של יחידת הרדאר לגלות רכב לפנים פוחתת במידה רבה אם המהירות של הרכב שלפנים שונה במידה רבה מהמהירות של המכונית שלך.
* שדה ראייה מוגבל-במצבים מסוימים כלי רכב אחר אינו מזוהה, או שהזיהוי נעשה מאוחר מהצפוי.
* במקרה של גשם כבד, או שלג או קרח על הסמל, ייתכן שתהיה הפחתה בתפקודי יחידת הרדאר, הם יושבתו כליל או יספקו תגובת תפקוד שגויה.

***תיאור הפרויקט:***

* **מבנה הפרויקט:**

הפרויקט נכתב בשפות:

python 3.9 המתאימה במיוחד לאלגוריתמים של למידת מכונה כמו yolo, OpenCV, coco***.***

C++ לכתיבת האלגוריתם.

כמו כן מורץ על סביבת Anaconda דרך Visual code .

* **קלט מהמשתמש:**

אני קולטת מהמשתמש את נקודת היעד שאליה הוא מעוניין לנסוע.

* **חיישנים:**

בפרויקט שלי נשתמש בחיישנים הבאים:

**לידאר**- לזיהוי המרחק של האובייקט מהרכב.

לידאר מזהה מרחק מאובייקטים באמצעות גלי אור.

**מצלמה**- על הרכב יותקנו מצלמות ב-4 כיוונים.

קדימה-כדי לראות כל הזמן את המרחב מלפני הרכב.

אחורה- כדי לראות כל הזמן את המרחב מאחורי הרכב.

מימין- כדי לראות כל הזמן את המרחב הימני של הרכב.

משמאל- כדי לראות כל הזמן את המרחב השמאלי של הרכב.

המצלמות יהיו מיוחדות עבור רכב אוטונומי- שמצלמות גם בלילה ע"י שימוש באינפרה אדום.

**IMU** – מכיל מד תאוצה, גירוסקופ-מודד מהירות זוויתית, שהיא קצב השינוי של מיקום או סיבוב זוויתי. הוא מזהה תנועות סיבוב סביב שלושת הצירים ומספק נתונים על קצב הסיבוב של האובייקט, כך נוכל לנווט את הרכב במהירות המותרת עפ"י חוקי התנועה וכן לדעת לאיזה כיוון הרכב נוסע.

**GPS** – חיישן לאיתור המיקום הנוכחי של הרכב. חיישן זה מדויק רק ב-96%.

* הפרויקט יעשה תוך הנחה מראש של חוסר התחשבות בבלתמי"ם-מצבים בלתי מזוהים ובלתי צפויים.

***בעיה אלגוריתמית:***

* **פתיחת פרויקט:**

יצירת פרויקט חדש בAnaconda – עם Python 3.9.

התקנת כל הספריות הדרושות לפרויקט.

התקנת כל התוספים למחשב כגון:Cuda, Pytorch..

* **רקע**:

כדי לקבל את כל הרקע על מכוניות אוטונומיות: איך הם בנויות, מאיזה חיישנים הם מורכבות, איזה אלגוריתמים מפעילים אותם אני אעשה קורס ב-coursera על מכוניות אוטונומיות וכל המידע הדרוש לי עליהם.

* **קבלת מידע מהחיישנים:**

**מידע על הסביבה-**

כדי לקבל מידע שוטף ועדכני על כל המרחב סביב הרכב האוטונומי והעצמים הנמצאים בו, נצטרך חיישן שבעצם יחקה את הראיה האנושית של הנהג. לצורך כך נתקין מצלמות ב-4 כיוונים: קדימה, אחורה ומשני הצדדים. שקולטות למרחק של עד 10 מטר.

כדי לדעת מה קורה בתמונות ולפעול בהתאם ננתח את המרחב באמצעות כמה סוגים של אלגוריתמים וספריות:

אלגוריתם yolo- שמתאים במיוחד לניתוח תמונות עבור רכב אוטונומי.

בחרתי אתv5 yolo כיוון שיש לו תוצאות מדויקות יותר וטובות יותר מ- yolo v3.

אלגוריתם yolo הוא אלגוריתם שמנתח תמונות בזמן אמת.

לאלגוריתם מכניסים dataset של חפצים/תמונות/דברים שמעוניינים לזהות אותם מתוך התמונה בזמן אמת.

מערכי נתונים ניתן להוריד מאתר Kaggle או ממקומות אחרים.

עבור המכונית לפרויקט שלי בחרתי לזהות באמצעות yolo :

רמזורים- רמזור אדום, רמזור ירוק, תמרורי כיוון מעל הרמזור שאומרים לאן עלי לפנות.

תמרור עצור- שמסמן לרכב לעצור.

לאחר הכנסת מערך הנתונים, מתחיל שלב האימון של האלגוריתם-train.

מגדירים את קטגוריות הדברים שברצוננו לזהות, לדוגמא: רמזור אדום ימינה-1, רמזור אדום שמאלה-2 וכו'... ומתחילים להריץ.

באמצעות תיבות סימון בצבעים שונים המתאימים לכל קטגוריה, נסמן כל פעם את אזור החפץ בתמונה. ככל שנאמן את האלגוריתם על מגוון גדול יותר של תמונות הוא ינפיק לנו בזמן אמת את התוצאות הנכונות ביותר עם פחות טעויות בזיהוי.

לאחר מכן עושים לאלגוריתם test-מבחן ובודקים מה רמת הדיוק שלו.

עבור מכונית חשוב שיהיו לנו תוצאות מדויקות ביותר כדי שלא יגרמו תאונות, ולכן יש לאמן אותו עד לדיוק מרבי.

בשביל ביצוע שלב זה יש ללמוד היטב את אלגוריתם yolo v5 ואת דרכי העבודה איתו.

כמו כן חיפוש מערך נתונים המכיל כמות נכבדת של תמונות העונות על הקריטריונים הנ"ל.

ספריית -OpenCV Opencv (ספריית ראיית מחשב בקוד פתוח) היא ספריית תוכנת ראייה ממוחשבת ולימוד מכונה בקוד פתוח שפותחה ע"י .intelהיא מספקת מגוון רחב של פונקציות ואלגוריתמים לעיבוד תמונה ווידאו, זיהוי תכונות, זיהוי אובייקטים ועוד.

לספריה זו פונקציות מיוחדות ביניהם פונקציה שיכולה לקלוט צבעים מסוימים מתמונה ולסמן את האזור בצבע. נעשה שימוש בפונקציה זו כדי לזהות את הנתיב הנוכחי שהרכב נוסע בו. זה יעיל ביותר כיוון שלמחשב קל לקלוט את שילוב הצבעים של הנתיב-שחור לבן וע"י סינון הרעש מתמונה- כל מה שלא עונה על הצבע הזה, ועוד כמה פעולות נוכל לקבל את אזור הנתיב באופן ממש מדויק.

לצורך זה יש לכתוב אלגוריתם לזיהוי נתיב המשתמש בספריה זו, ולאמן אותו על תמונות של נתיבים ולבדוק שזה אכן מזהה בצורה מדויקת וטובה.

ספריית COCO- ספריית COCO (Common Objects in Context) היא מערך נתונים ומדד הערכה בשימוש נרחב עבור משימות זיהוי, פילוח וכתוביות של אובייקטים בראייה ממוחשבת. הוא מספק מערך נתונים בקנה מידה גדול עם תמונות מוערות, תיבות תוחמות אובייקט, מסכות פילוח וכיתובים. מערך הנתונים של COCO מכיל קבוצה מגוונת של אובייקטים נפוצים הנמצאים בסצנות יומיומיות, מה שהופך אותו למתאים לאימון והערכת מודלים של ראייה ממוחשבת. ספריית COCO כוללת גם מדד הערכה הנקרא ,COCO AP (Average Precision) המודד את הדיוק של אלגוריתמי זיהוי אובייקטים ופילוח. מדד זה לוקח בחשבון ערכי דיוק וזיכרון כדי לספק הערכה מקיפה של ביצועי המודל.

ספריית COCO כבר מאומנת בזיהוי של אנשים, רכבים ואופנועים ולכן אני אעדיף להשתמש בה.

מה שנותר לעשות הוא לחפש קוד שמזהה רכבים, אופנועים ואנשים באמצעות ספריית COCO ולהטמיע אותו בפרויקט.

כאשר נרצה לקבל מידע על סביבת הרכב בצד מסוים, נגש למצלמה שנמצאת בצד הרצוי על הרכב, ננתח את הנתונים באמצעות האלגוריתמים שהזכרנו, והרי לנו מפה מדויקת של סביבת הרכב.

**מרחק**-

כדי לדעת מה המרחק של הרכב מהרכבים שבסביבת הרכב או ממכשולים נשתמש בחיישן לידאר שמחזיר לנו את המרחק של העצם מהרכב.

חיישן זה שולח גלי אור ומחזיר ענן נקודות של הסביבה.

אפשר לנתח ענן זה ע"י אלגוריתמים שונים כמו: פילוח ענן נקודות ולהגיע למיפוי כולל של הסביבה.

אני מעדיפה לזהות את העצמים בסביבה דרך מצלמות, ולהשתמש בלידאר לזיהוי מרחק. עבור כל נקודה מהענן נקודות שמחזיר הלידאר יש לנו 3 פרמטרים:

המרחק של העצם מהמכונית והמיקום בציר ה-X ובציר ה-Y קואורדינטות קווי הרוחב והאורך של העולם.

באמצעות מהירות האור הידועה, שהיא קבועה, חיישן הלידר מחשב את המרחק לעצם על ידי הכפלת זמן הטיסה במהירות האור ולאחר מכן חלוקתו בשניים (מאז הדופק נוסע אל האובייקט ובחזרה).

לצורך כך יש ללמוד התעסקות בסיסית עם ענן נקודות שפולט לידאר,

וכן לכתוב קוד שמקבל את הנתונים מהחיישן ומחזיר לי את המרחק מהעצם לרכב.

**מהירות**-

נתקין על הגלגלים **IMU** - הפלט של IMU הוא מהירות נוכחית של הגלגל בזווית מסויימת. וכך נוכל לדעת ע"י מספר חישובים, בכל זמן נתון מה המהירות הנוכחית של הרכב, וכן האם צריך להאט או להגביר את המהירות.

חקירה איך עובד מד מהירות ומה הפלט שלו.

כתיבת קוד שהפלט מקבל את הפלט של מהירות זוויתית של הגלגל ומחזיר מהירות נוכחית של הרכב.

**מיקום וניווט**-

זיהוי מיקום נוכחי של הרכב יעשה באמצעות חיישני GPS.

חיישן GPS מספק מידע מיקום גיאוגרפי מדויק. הוא משתמש באותות מלוויינים מרובים כדי לקבוע את קואורדינטות קווי הרוחב והאורך של המכונית.

ניתן להשתמש במידע זה כדי לאתר את המיקום המדויק על פני כדור הארץ.

באמצעות אלגוריתם מסנן קלמן נוכל לשערך את המיקום הנוכחי בצורה אופטימלית של הרכב.

כמו כן כדי לדעת לאן לנווט את הרכב נצטרך לעשות שימוש במידע שמגיע מספריות הGoogle Maps ולדעת בכל שלב לאן הרכב צריך להמשיך.

* **חישוב מסלול:**

בחלק זה יכנס גם הצד לקוח של המשתמש.

למשתמש יהיה מסך שבו הוא יקליד את כתובת היעד שהוא מעוניין לנסוע אליו.

נעשה זאת באמצעות התחברות לGoogle Maps.

Google מאפשרת לנו שימוש בספריית Google Maps שמכילה את כל מפות הדרכים של הכבישים בעולם.

כמו כן מציעה מגוון רחב של ספריות שונות שמציעות ממשק משתמש אטרקטיבי ומנגד גם ספריות שמסייעות למתכנת-כמו חישוב המסלול הקצר ביותר מכתובת המוצא לכתובת היעד תוך התחשבות בגורמים משתנים כמו: עמס בכבישים, תאונות, חסימות כבישים וכו', זה נעשה ע"י שימוש באלגוריתמים שונים של גרפים כמו דייקסטרה וכו'...

יש להשיג מפתח API זהו מפתח שע"י ניתן להתחבר ל Google Mapsדרך קוד.

זה יהיה פרויקט מסוג- WEB כיוון שזה ממשק משתמש.

כמו כן ללמוד הייטב את ספריות Google Maps.

לאחר מכן-כתיבת קוד שקולט מהמשתמש את נקודת היעד, ניתן לו גם אפשרות של השלמה אוטומטית ותיבת חיפוש, מזהה מיקום נוכחי של הרכב באמצעות חיישני GPS, ומחב את המסלול הקצר ביותר לכתובת היעד.

בנוסף מציג על המסך את מפת הדרכים ומסמן את המסלול שעל הרכב לנסוע.

* **היתוך חיישנים:**

היתוך חיישנים ברכב אוטונומי מתייחס לתהליך של שילוב נתונים ממספר חיישנים כדי לקבל הבנה מדויקת ואמינה יותר של סביבת הרכב. בכלי רכב אוטונומיים מותקנים בדרך כלל חיישנים שונים, כגון מצלמות, מכ"ם, לידר וחיישנים קוליים, כל אחד מספק סוגים שונים של מידע על הסביבה.  
אלגוריתמי היתוך חיישנים לוקחים את הנתונים הגולמיים מחיישנים אלו ומשלבים אותם יחד כדי ליצור תפיסה מקיפה והוליסטית של הסביבה. על ידי שילוב החוזקות של חיישנים שונים ופיצוי על המגבלות האישיות שלהם, היתוך חיישנים משפר את יכולות התפיסה של הרכב ומאפשר קבלת החלטות חזקה יותר.  
תהליך ההיתוך כולל יישור וסנכרון של נתוני החיישן, הסרת רעשים וחריגים, כיול מדידות חיישנים, ולאחר מכן איחוי הנתונים יחד באמצעות טכניקות שונות כמו סינון, מעקב אחר אובייקטים, מיפוי ושיטות אומדן הסתברותיות.  
נתוני החיישן המותכים המתקבלים מספקים ייצוג מדויק יותר של הסביבה, ומאפשרים לרכב האוטונומי לתפוס טוב יותר עצמים, לזהות מכשולים, לעקוב אחר עצמים נעים, להעריך את מיקומם ומהירותם ולקבל החלטות מושכלות לגבי ניווט, תכנון נתיב, הימנעות מהתנגשות ופונקציות נהיגה אוטונומית אחרות. היתוך חיישנים ממלא תפקיד קריטי בשיפור האמינות הכוללת, הבטיחות והביצועים של כלי רכב אוטונומיים.

החלטתי לעשות את היתוך החיישנים באמצעות אלגוריתם מסנן קלמן מורחב-EKF.

מסנן קלמן זהו אלגוריתם שמקבל מידע מהחיישנים של המכונית כגון: מיקום ומהירות, אך כיוון שהמידע שמגיע לא נכון במאה אחוז, וכן לכל חיישן יש מצבים מסוימים שבהם הוא לא מספיק מדויק, או לא עובד בכלל, יש צורך לבצע חישובים ע"י סטטיסטיקה ופונקציות של מודל התנועה וע"י סנכרון המידע נוכל לשערך בצורה הנכונה ביותר הנתונים מהחיישנים וכך לנווט את הרכב בצורה הטובה ביותר. הוא מתבסס על הרעיון של הסתברות בייס- שבהינתן מאורע וסיגמא ניתן לחשב מאורעות נוספים.

לביצוע שלב זה:

למידת קורס אלגוריתמי ניווט ושיערוך מקום.

חקירה יסודית של מסנן קלמן.

כתיבת פונקציית מסנן קלמן רגילה. שדרוג הפונקציה להיתוך חיישנים. בדיקה שהפונקציה עובדת הייטב. פונקציה זאת היא בעצם הפונקציה הבסיסית והכי חשובה של הרכב.

בכל זמן נתון הפונקציה מאגדת לנו את כל המידע הנדרש, ע"י כיול והיתוך החיישנים

וכך נוכל לשלוף ממנה את המידע שאנחנו צריכים בצורה נוחה ונגישה.

* **פונקציית ההפעלה:**

פונקציית ההפעלה מאגדת את כל האלגוריתמים השונים והמידע העצום שמגיע מהחיישנים.

היא בעצם "המוח" של הרכב, במקום המוח של הנהג.

פונקציה זו כוללת בתוכה את החלטות הנהיגה- מה עושים בכל מקרה, מתי לנסוע ומתי לעצור.

היא תהיה מחולקת כך:

ניווט: פונקציית ההפעלה צריכה לכלול אלגוריתמים לתכנון וביצוע מסלול המכונית.

זה כרוך בהתחשב בגורמים כמו יעד, תנאי תנועה, חוקי כביש וכלי רכב אחרים.

קביעת המיקום המדויק של הרכב- יעשה באמצעות מסנן קלמן, קבלת מידע מחיישן-GPS ו-Google Map.

תפיסה: תכלול את המידע המגיע מהאלגוריתמים למשימות תפיסה, כגון זיהוי אובייקטים, זיהוי נתיב, זיהוי כלי רכב, זיהוי תמרורים וזיהוי הולכי רגל. זה מאפשר למכונית להבין את סביבתה ולקבל החלטות מושכלות.

יעשה באמצעות מסנן EKF וכתיבת פונקציות בשפת C++.

בקרה: פונקציית הבקרה צריכה לכלול אלגוריתמי בקרה לטיפול בתאוצה, בלימה והיגוי של המכונית. אלגוריתמים אלו מבטיחים שהמכונית מתמרנת בצורה בטוחה וחלקה בהתבסס על המידע המגיע מפונקציית התפיסה והמסלול המתוכנן. כתיבת פונקציות בשפת C++.

קבלת החלטות: פונקציית ההפעלה צריכה לכלול אלגוריתמי קבלת החלטות הלוקחים בחשבון גורמים שונים, כמו מצב המכונית הנוכחית, תנאי הדרך, חוקי התנועה והתנהגות משתמשי דרך אחרים. אלגוריתמים אלו קובעים את פעולות המכונית, כגון החלפת נתיב, עקיפה או עצירה בצמתים.

באמצעות מסנן EKF נוכל לדעת את המידע המדויק באותו זמן על החיישנים- כי הוא עושה היתוך חיישנים.

ניצור טבלת SQL. לטבלה נכניס בכל פעם את מה שהתקבל מפונקציית התפיסה: הולך רגל בכביש, רמזור אדום, לפנות ימינה וכו', עפ"י עדיפות מסוימת,

הפונקציה תשלוף ממסד הנתונים את שם הפונקציה שצריך לבצע במקרה הנ"ל,

או שיש צורך לעבור לבדיקות נוספות במסד נתונים אחר ותבצע אותם.

טבלאות הdata base – ינהלו את סדר העדיפות ודרכי הפעולה של הרכב.

***תהליכים עיקריים בפרויקט:***

מקבלים מהמשתמש את נקודת היעד אליה הוא רוצה להגיע.

מחשבים את המסלול הקצר ביותר לנקודת היעד.

מתחילים לנסוע.

במשך הנסיעה- קולטים עם החיישנים מידע על הסביבה, עצמים, מרחק, מהירות וכו'...

ומעבדים אותו באמצעות האלגוריתמים השונים.

מעבירים את המידע לפונקציית קבלת ההחלטות שמחזירה לנו את שם הפונקציה שאנחנו צריכים להפעיל כדי שהרכב יתפקד כנדרש.

מפעילים את הפונקציה.

מבצעים את השלבים שוב ושוב עד שמגיעים אל היעד!

***הסתייגויות*:**

* הפרויקט עובד על כבישים חד סיטריים או עם 2 נתיבים בלבד.
* הפרוייקט לא עוסק בהתעסקות עם מצבי בלתמ"ים.
* כאשר אני צריכה לקבל מידע מלידאר- אני מפעילה אותו למיקום ספציפי והוא מחזיר לי את המרחק. אני מניחה שאני יכולה להפעיל למיקום מסויים וכן שלא יופיע לי במהלך המדידה אובייקט שלא צפיתי.
* כאשר אני מזהה רמזור אני מניחה שאני צריכה להאט ולעצור.
* המכונית לא תעבוד במזג אוויר קיצוני.
* לא התייחסתי לכיכרות.
* זיהוי הנתיב לא יעבוד על כבישים מפותלים במיוחד.