

## תדריך: מכונית אוטונומית

כללי: במעבדה זאת תיצרו שני יישומים, בראשון הרובולגו ינווט את עצמו אוטונומית לאובייקט שהשתמש יבחר, בשני הרובולגו ינווט את עצמו אוטונומית לעבר המשתמש (או תמונה שלו). היישום השני הנו רשות

### חבילות חדשות:

- [CvLib](#).
- [Face recognition](#). (ליישום השני).

### תשתית:

- מכונית אוטונומית: רובולגו ועליו מורכב טבלט.
- חיישן המכונית האוטונומית: מצלמת הטבלט.
- אובייקטים: אמתיים או תמונות.
- תקשורת: רשת פנימית gadwl תחת חלונות, ו-nat או bridged\_adapter במכונה הווירטואלית של הלינוקס.

### עזרה לשני היישומים:

- קובץ עזרה למחשב קרוב במודל לשני היישומים.
- קובץ עזרה למחשב רחוק במודל לשני היישומים.
- סרט הסבר מבוא לשני היישומים.
- סרט הסבר למחשב הקרוב לשני היישומים.

### המחשב הקרוב בשני היישומים:

- דומה מאוד ל-IOT.
- בניסוי וטעיה נמצא שהרובולגו נתקע בשלב של סריקה תוך כדי סיבוב ימינה ושמאלה, להתגבר על הבעיה, לכל אינדיקציה של נסיעה קדימה הוספה השהייה של שנייה, והסיבובים נעשים בעזרת כיוונים מנוגדים של המנועים בהשהייה קצרה יחסית.

### יישום ראשון:

#### עזרה ליישום הראשון:

- הקדמה סרט יישום ראשון.
- סרט מחשב רחוק ליישום ראשון.

### המחשב הרחוק:

- התקשורת מתבצעת לשרת רדיס יחיד, שכולם יכתבו ויקראו אליו.
- כל משתמש מזהה ע"י שם הרובולגו.
- יש להניח את "המכונית" על רצפת המעבדה ולכתוב קוד כך שהטבלט יעביר למחשב את התמונה (ראה קוד).
- עליך להשתמש בחבילת cvlib כך שתתחיל לזהות אובייקטים. (ראה קוד).

- אם אובייקט המטרה שברצונך להגיע אליו לא מזוהה יש ליצור אפשרות לנהג את המכונית דינית באמצעות מקשי החיצים של המקלדת, עד להגעה למרחק מספיק קרוב כך שאובייקט המטרה מתחיל להיות מזוהה.
- ברגע שאובייקט המטרה מזוהה עליך למקם את חץ העכבר במרכז המטרה וללחוץ על הלחצן השמאלי, אם הכל תקין תוכל לראות בנוסף לשם האובייקט גם את שם הרובולוג שלך, ובנוסף תוכל לראות את כל שאר שמות הרובולוגים שעשו תהליך דומה.
- יש להעביר לרדיס את שם הרובולוג שלך ושם המטרה לדוגמה אם ברצונך שהמכונית תנווט אוטונומית לכסא ושם הלוג שלך הינו dan, העבר `dan:target` כמפתח ואת שם האובייקט כערך, בפייטון לדוגמה `r.set('dan:target','chair')`.
- לאחר "רכישת" המטרה יש לכוון את מרכז המטרה למרכז המסך, אם מרכז המטרה ימינה ממרכז המסך, יש להפנות את הרכב שמאלה ולהפך.
- בהגעה למטרה, או שחיישן המגע נלחץ ומפסיק את הרצת היישום או המשתמש לוחץ על המקש Q והיישום מפסיק, הרובולוג מקבל הוראה לעצור, ויש למחוק את כל המפתחות והערכים של המשתמש ברדיס.

יישום שני (רשות):

עזרה ליישום השני:

- [הקדמה סרט יישום שני.](#)
- [סרט מחשב רחוק ליישום השני.](#)

המחשב הרחוק:

- התקשורת מתבצעת לשרת רדיס יחיד, שכולם יכתבו ויקראו אליו.
- כל משתמש מזוהה ע"י שם הרובולוג.
- "המכונית" מונחת על רצפת המעבדה והטבלט מעביר למחשב את התמונה ([ראה קוד](#)).
- כל משתמש יצלם את עצמו ויכתוב את הנתביב לתמונה בקוד, שם קובץ התמונה שלו יישמש כתווית לתמונה ויועבר לרדיס, לדוגמה אם הרובולוג שלי שמו bar אעביר לרדיס תחת המפתח `bar:label` את הערך `gad_halevy` שמשמש גם כשם לתמונה. כל משתמש גם מעביר את ה-`encodings` של התמונה כערך תחת המפתח בדוגמה `bar:encodings`.
- כל משתמש קורא מרדיס את התוויות של התמונות ואת ה-`encodings` שלהן כמחרוזות.
- התכנה מתרגמת את המחרוזות ל-`encodings` ומשייכת ע"פ ה-`encodings` את התוויות המתאימה לכל תמונה.
- התכנה מזזה את תמונת המשתמש (או את פני המשתמש עצמו) ומנווטת את הרכב האוטונומי אליו.
- בהגעה למטרה, או שחיישן המגע נלחץ ומפסיק את הרצת היישום או המשתמש לוחץ על המקש Q והיישום מפסיק, הרובולוג מקבל הוראה לעצור, וכל המפתחות והערכים של המשתמש ברדיס נמחקים.

להגשה בוריפייר:

- סרט של הרצת המשימה הראשונה שבו יראו את המסך של המחשב הקרוב עם שם הרובולוג והתאריך, שבו יראו הן את האובייקט הנבחר על המסך יחד עם שם הרובולוג שלכם, ובו יראו גם את הרובולוג נוסע לכיוון המטרה. בקיצור עליכם לעשות סרט שיוכיח חד משמעית שאתם כתבתם את האפליקציה והרצתם אותה.
- קוד.

נערך ע"י גד הלוי 01/24 © כל הזכויות שמורות לאוניברסיטת ת"א.  
 אין להשתמש בתדריך זה ואו בחלקים ממנו ללא אישור המחבר.  
 דף 2 מתוך 3



# בהצלחה!