

ראיה ממוחשבת 0510.6251

סמסטר ב' תשע"ח

נהלי הפרויקט

1. הפרויקט הינו בזוגות.
2. הציונים יקבעו על בסיס תחרות ביצועים, על פי המדד בסעיף 13.
3. הפרויקט עוסק בגילוי וסיווג אוטובוסים. בעזרת תמונות האימון שאותם ניתן להוריד דרך המודל, תצטרכו לפתח אלגוריתם שמגלה ומסווג אוטובוסים. לכל זוג בסוף הסמסטר תינתן רבע שעה על מנת להסביר בקצרה איך האלגוריתם עובד ולבדוק איך הוא מגלה תמונות של אוטובוסים שהאלגוריתם לא ראה מעולם.
4. את הפרויקט יש לכתוב בשפת python. מומלץ להתקין anaconda (גרסה 5.1.0, python 3.6), ניתן להתקין דרך [האתר הרשמי](#).
5. מומלץ להכין את הפרויקט בעזרת כלים של למידה עמוקה, אך זה לא חובה. במידה ואכן נעשה שימוש ברשתות נוירונים, מומלץ להשתמש באחת מ-2 החבילות הבאות:
 - TensorFlow (version 1.6.0) (ניתן להריץ pip install tensorflow ב-command line).
 - pyTorch (version 0.3.1) ([לינק להתקנה](#)).בנוסף ניתן לעשות שימוש גם ב- openCV (version 3.4.0.12).
- מומלץ להשתמש בגרסאות ה"ל על מנת שתהיה התאמה עם הגרסאות במחשב הבדיקה ושלא תהיינה בעיות בלתי צפויות.
- קובץ עם כל החבילות שמותקנות במחשב הבדיקה נמצא במודל תחת השם requirements.txt. במידה ותרצו להשתמש בחבילות נוספות **אנא ידעו אותי** על מנת שניתן יהיה להתכונן לכך מראש.
6. המטרה היא למצוא איפה נמצא כל אוטובוס בתמונה ובנוסף באיזה אוטובוס מדובר (יש סה"כ 6 אוטובוסים שונים). הגילוי של האוטובוס יעשה על ידי המרובע החוסם המינימאלי שמכיל את כל האוטובוס (דוגמאות בהמשך). המרובע יוגדר על ידי 4 פרמטרים, xmin, ymin, width, height. כאשר xmin, ymin הם הקואורדינטות של הפינה השמאלית העליונה של המרובע, width הוא הרוחב ו-height הוא הגובה של המרובע. בנוסף, מספר בין 1 ל-6 יגדיר את סוג האוטובוס, כאשר {ירוק=1, צהוב=2, לבן=3, אפור=4, כחול=5, אדום=6}. לסיכום, לכל גילוי יש 5 פרמטרים, ארבעת הפרמטרים הראשונים מגדירים את המיקום של המרובע (גילוי) והפרמטר החמישי מגדיר את סוג האוטובוס (סיווג). המוצא של האלגוריתם צריך להיות קובץ טקסט כך שכל שורה מגדירה תמונה אחת כאשר הפרמטרים מופרדים בפסיקים.
PIC.JPG:[xmin1, ymin1, width1, height1,color1],...[xminN, yminN, widthN, heightN,colorN]

לדוגמא:

DSCF1013.JPG:[1217,1690,489,201,1],[1774,1619,475,224,2]

DSCF1015.JPG:[641,1342,1181,892,3]

DSCF1016.JPG:[1067,1843,1114,613,4],[1954,1278,1021,561,6]

בדוגמא הנ"ל יש שלוש תמונות, בתמונות DSCF1016, DSCF1013 וזהו 2 אוטובוסים ובתמונה DSCF1015 וזהו אוטובוס אחד.

7. קובץ בפורמט הנ"ל עם תיוג האמת נמצא במודל לשימושכם.

8. הגדרה של גילוי נכון:

גילוי אמת יוגדר על ידי יחס חפיפה IOU (intersection over union) בין המרובע החוסם של הגילוי לבין המרובע החוסם של תיוג האמת. יחס החפיפה מוגדר על ידי:

A – detection bounding box

B – ground truth box

$$IOU = \frac{A \cap B}{A \cup B}$$

גילוי יחשב כגילוי נכון בתנאי ש-2 התנאים הבאים מתקיימים:

- $IOU > 0.7$

- הסוג של האוטובוס הוא הסוג הנכון (הצבע).

לבסוף מחושב מדד [F1Score](#), זהו המדד היחיד הממשקל את כל תוצאות הגילוי מכל התמונות, זהו המדד שבעזרתו יבחן טיב הפרויקט ביחס לפרויקטים אחרים. (יש גם משקל לזמן הריצה).

9. סקריפט הגילוי שנדרש להכין:

עליכם לייצר סקריפט בשם runMe.py, הסקריפט צריך לכלול פונקציה בשם run שתקבל כקלט את שם קובץ התיוגים שהסקריפט ייצר ואת התיקיה שבה נמצאים כל התמונות שאותן צריך לתייג. כלומר ההגדרה של הפונקציה צריכה להיות מן הצורה:

```
def run(estimatedAnnFileName, busDir):
```

חשוב להשתמש בשמות הנ"ל על מנת שסקריפט הבדיקה (מוסבר בהמשך) יעבוד ללא תקלות.

10. במודל, בתוך הקובץ python_scripts.rar, נמצאים 3 קבצי פייתון, Main.py, runMe.py,

busProjectTest.py.

runMe – זהו קובץ שמייצר קובץ תיוגים על ידי הרעשה של הקובץ המקורי. זהו הקובץ שתצטרכו להחליף בקובץ שלכם שיכין את תיוג האמת. ניתן להריץ אותו על מנת לבדוק שהפייתון עובד כראוי, על מנת להריץ אותו תדרשו לשמור את קובץ תיוג האמת בתיקיה שממנו קוראים לסקריפט. busProjectTest – סקריפט שמשווה בין תיוג האמת לתיוגים המשוערכים. Main – הסקריפט המרכזי שרק אותו נדרש להריץ. זהו הסקריפט שבעזרתו יבדק הפרויקט על תמונות המבחן, לכן חשוב לבדוק את אלגוריתם הזיהוי בעזרת הסקריפט הנ"ל.

על מנת לבדוק שאכן סקריפט הבדיקה עובד כראוי ניתן להוריד את שלושת הקבצים לאותה תיקיה ולהריץ את קובץ ה-Main, קובץ זה קורא ל-2 הקבצים האחרים, עבור כל תמונה מחשב את מספר

תיוג האמת (TP) מספר הפספוסים (Missed) ומספר תיוג השקר (FP), לאחר מכן מחשב את הציון הכללי ושומר הכל.

הרצה של ה-Main:

Main מקבל כקלט את קובץ תיוג האמת (מה שמורידים מהמודל), את שם הקובץ של התיוגים המשווערכים (מה ש-runMe ייצר), את התיקיה שבה נמצאים כל התמונות ואת התיקיה שבה הסקריפט ישמור את כל התוצאות. לפני כל קלט נדרש לכתוב באיזה קלט מדובר כאשר:

```
-myAnns <estimated annotations file>
-anns <real annotations file>
-buses <directory of all the images>
-saveDir <directory for the output of the script>
```

את הסקריפט ניתן להריץ דרך ה-command line, דוגמא להרצה של הסקריפט:

```
python Main.py -myAnns newAnns.txt -anns realAnns.txt -buses busesDir
-saveDir dirToSave
```

בנוסף על מנת לקבל הסבר על כל קלט ניתן לקרוא לסקריפט עם -h כלומר:

```
python Main.py -h
```

11. הגשת הפרויקט:

- כל זוג צריך להגיש במועד שנקבע להגשת הפרויקט:
- דו"ח בהיקף של 2-3 עמודים המסביר את הפתרון שמומש.
- את כל הקבצים הנדרשים על מנת להריץ את אלגוריתם הגילוי (עדיף כקובץ zip).

את כל הנ"ל יש לשלוח למייל pihash@gmail.com עם הכותרת

cv_proj_18_ID1_ID2

במקום ID1, ID2 יש לכתוב את ת.ז של המגישים.

12. מהלך הבחינה:

- הסבר קצר של הסטודנטים על האלגוריתם שמומש.
- הרצת האלגוריתם על תמונות המבחן וקבלת F1 Score.
- מומלץ להגיע גם עם "דיסק און קי" המכיל את כל הקבצים הדרושים להרצה של האלגוריתם
- למקרה שיהיו בעיות כלשהן עם הקבצים שנשלחו במייל.

13. ציון:

הציון הסופי של הפרויקט יכול להיות ציון על ביצועי הגילוי במשקל של 75% וציון על ביצועי זמן במשקל של 25%. כאמור, הציון הסופי של הפרויקט יקבע על ידי תחרות בין כל הפרויקטים.

דוגמאות לפלט של סקריפט הבדיקה (Main.py):



