

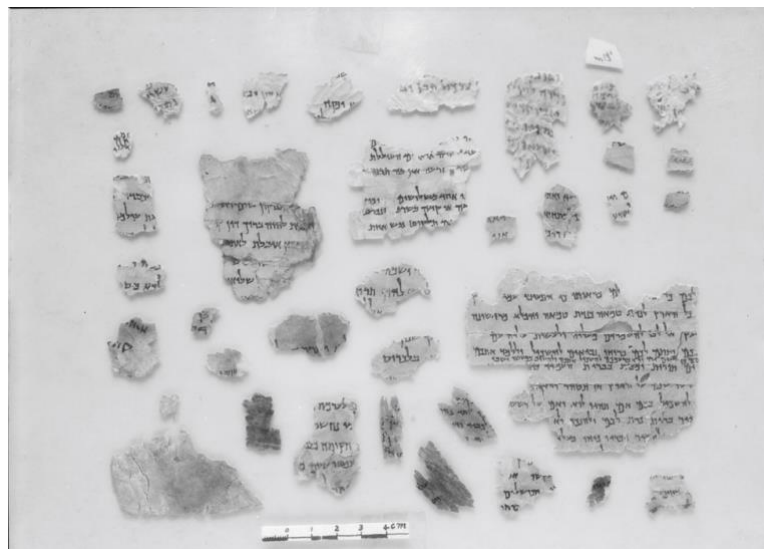
פרויקט סיום בקורס רשתות נירונים עמוקות - 2025 הגשה עד 14.3.2025 בשעה 12:00

מי שגיש עד ה 25.2.2025 יקבל בونוס של 4 נק'

מי שגיש עד ה 3.3.2025 יקבל בונוס של 2 נק'

תיבות חוסמות של צילומי קטעי מגילות עתיקים מדבר יהודה

דוגמא לנתונים בפרויקט



מטרת הפרויקט הינה אימון רשתות אשר יזהו את קטעי המגילות

https://www.dropbox.com/scl/fo/9j9730xksxfzg7cnxyp9/AHfzSUHTISv_VyjdD3Sjdl=0&_JQ?rlkey=y2sxyb82obhqexrjo8hl9eh0q

התמונות ניתנות לכם לשימוש אקדמי בלבד! אין לשמור את התמונות אצלכם לאחר סיום הפרויקט! ואין לעשות בהם כל שימוש מלבד בתרגיל זה.

מבוא

כל סטודנט יקבל חלק אחר של התמונות לאימון ולבדיקה – הקישור לתמונות יישלח אליכם בנפרד.

אתם נדרשים להכין את הנתונים לאימון. יש להשתמש אך ורק בתוכנת LabelMe (<https://github.com/wkentaro/labelme>) לצורך הסימון! התוכנה חייבת להיות מותקנת מקומית, אין להעלות את הנתונים לשום שירות אחר.

קובציה לאופן אופן מיספור הקטעים בפרויקט – מספר הקטע יהיה המרחק בין הפינה השמאלית העליונה של המלבן המקיף את המגילה לפינה העליונה של התמונה כפי שרואים בדוגמא שלהלן.



חלק א' - סימון ראשי ואימון והערכת איכות (40%)

בשלב הראשון, עליכם כאומר לבצע סימון ראשוני של המגילות בתמונה. יש לבצע זאת באמצעות תיבות חוסמות (Bounding Boxes) המקיפות כל מקטע מגילה בתמונה. יש לשים לב כי חלק מהתיבות עשויות להיות חופפות.

דרישות לביצוע:

1. לכל מקטע מגילה בתמונה יש ליצור תיבה חוסמת אדומה באמצעות תוכנת LabelMe.
2. יש למספר כל מקטע מגילה בתמונה. המספור יעשה כפי שהוסבר לעיל על פי המרחק בין הפינה השמאלית העליונה של התמונה לפינה השמאלית העליונה של התיבה החוסמת.
3. אימון רשת ע"פ בחירתכם. כולל כל השלבים שעשיתם בתרגיל הבית האחרון. יש לדווח על 3 הרשתות הטובות ביותר והיעילות ביותר. (עדיפות לאיכות הסימון על-פני יעילות כמובן)
4. עליכם לחשב ולדווח את מדד ה-Intersection Over Union -- IOU בין הסימון שלכם לבין סימון ייחוס שיסופק לכם.

בבדיקת התרגיל אנו נבדוק בעיקר את הפונקציה אחת בלבד עם החתימה הבאה. ולכן כאשר מרצים את הפונקציה הזאת מתוך תיקיית התרגיל שלכם התרגיל צריך לרוץ חלק ולספק את הפלט הרלוונטי. כלומר כל הנתונים שאתם מגשים צריכים להיות יחסיים למיקום של הפונקציה הזאת

```
def predict_process_bounding_boxes(image_path: str, output_csv: str) -> None:
    """
    Processes an image to detect bounding boxes around scroll segments.
    Saves the bounding box data to a CSV file.

    Args:
        image_path (str): Path to the input image. (full paths to the input images)
        output_csv (str): Path to the output CSV file.
    """
    pass
```

דוגמה לפלט קובץ CSV בחלק א':

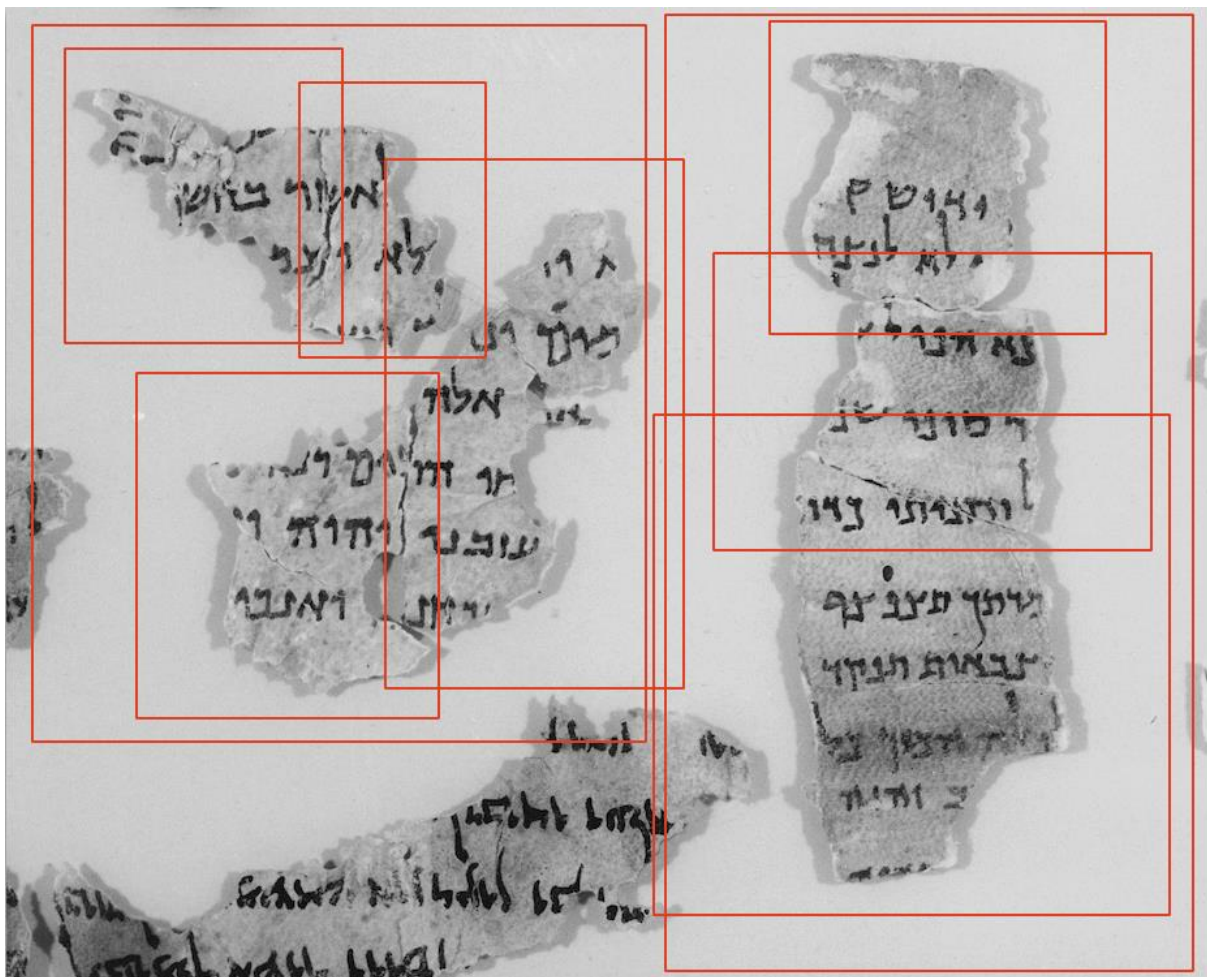
כמובן שהעמודה iou תהיה רק עבור ה-training set. עבור נתוני הבדיקה זה יהיה מינוס אחד

```
image_name,scroll_number,xmin,ymin,xmax,ymax,iou  
image1.jpg,1,30,40,200,250,0.85  
image1.jpg,1,50,60,220,270,0.90  
image2.jpg,2,20,30,180,230,0.88
```

חלק ב' - סימון מתקדם של מקטעים קטנים יותר (50%)

בחלק זה, עליכם לבצע סימון מתקדם יותר של המגילות בתמונה, תוך תשומת לב לכך שחלק מהמגילות מורכבות ממקטעים קטנים יותר אשר חוברו.

דוגמא לתתי מקטים



דרישות לביצוע:

1. יש ליצור גם תיבות חסומות לכל מקטע מגילה גדול וגם תיבות לכל אחד מהמקטעים הקטנים המרכיבים אותו.
2. גם כאן יש לבצע מספור של המקטעים לפי המרחק מפיינה שמאלית עליונה של התמונה.
3. הנתונים ייוצאו לקובץ CSV בפורמט דומה כמו בחלק הראשון

גם כאן. בבדיקת התרגיל אנו נבדוק פונקציה אחת בלבד עם החתימה הבאה. ולכן כאשר מרצים את הפונקציה הזאת מתוך תיקיית התרגיל שלכם התרגיל צריך לרוץ חלק ולספק את הפלט הרלוונטי. כלומר כל הנתונים שאתם מגשים צריכים להיות יחסיים למיקום של הפונקציה הזאת

```
def process_detailed_bounding_boxes(image_paths: list[str], output_csv: str) ->
list[str]:
    """
    Processes a list of image file paths to detect detailed bounding boxes
    for both large and small scroll segments.
    Saves the bounding box data to a CSV file.

    Args:
        image_paths (list[str]): List of full paths to the input images.
        output_csv (str): Path to the output CSV file.

    Returns:
        list[str]: List of full paths (path + file name) of processed images.
    """
    pass
```

דוגמה לפלט קובץ CSV בחלק ב':

כמובן שהעמודה iou תהיה רק עבור ה-training set. עבור נתוני הבדיקה זה יהיה מינוס אחד.

ללא iou

ה scroll_number – זה לפי הסדר מספור של המקטעים לפי המרחק מפיינה שמאלית עליונה של התמונה.

```
image_name,scroll_number,xmin,ymin,xmax,ymax,iou
image1.jpg,1,30,40,200,250,0.85
image1.jpg,1,50,60,220,270,0.90
image2.jpg,2,20,30,180,230,0.88
image2.jpg,2,25,35,150,200,0.92
```

נהלי הגשה (10% מהציון)

ההגשה כוללת:

1. דו"ח מפורט וברור הכולל:

- שם המגיש ומספר תעודת זהות.
- תיאור הפתרון המוצע לכל חלק.
- הסברים, צילומי מסך ותוצאות מספריות.
- הפניה לקבצי הקוד הרלוונטיים. (אפשר גם צילומי מסך של קטעי קוד עיקריים)
- כמו כן, יש לצרף את כל הנתונים הרלוונטיים והסברים מפורטים איך לאמן את ה

2. קבצי קוד ב-ZIP אחד המכיל:

- Readme.doc
- projectDL2025_q1.py – קובץ ראשי עבור חלק א'
- projectDL2025_q2.py – קובץ ראשי עבור חלק ב'
- תיקייה בשם images עם חמש התמונות שקיבלתם
- הקוד צריך לפעול כך שניתן יהיה להריץ את הקובץ הראשי וכל התמונות ייפתחו אוטומטית.

3. קישור לקובץ המשקולות בו השתמשתם לאימון המודל – ניתן להעלות את הקובץ ל- Google Drive או לכל שירות ענן אחר ולצרף קישור בגוף הדו"ח.

4. כמובן שאיכות הדיוק, איכות המימוש, יעילות יהיה חלק מהציון