

## פרויקט סטגנוגרפיה

### כללי

Steganography היא שיטה להסתרת מידע בתוך קובץ, הודעה, תמונה או וידאו. עבודה זו מתמקדת בהסתרת מידע בתוך תמונה. הסתרת המידע בתמונה מבוססת על שימוש בייצוג בינארי של תווים/מספרים לצורך שינוי ערכי הפיקסלים של התמונה.

### חלק א

#### המשימה

בהינתן קובץ טקסט ותמונה, עליכם לייצר תמונה חדשה בה הטקסט חבוי בביטים הנמוכים של כל פיקסל.

#### רקע

כל פיקסל בתמונה מורכב משלושה ערכים (RGB), כל אחד מהם מתורגם לוקטור בינארי של 8-ביטים. כל תו בטקסט מתורגם גם הוא לוקטור בינארי של 8-ביטים. בפעולת ההסתרה, מטמינים את ערכי הוקטורים הבינאריים של תווי הטקסט בתוך הוקטורים הבינאריים של ערכי הפיקסלים.

#### דוגמה

תמונה בגודל 4x3 מיוצגת על-ידי המערך הבא :

[(27, 64, 164), (248, 244, 194), (174, 246, 250), (149, 95, 232),  
(188, 156, 169), (71, 167, 127), (132, 173, 97), (113, 69, 206),  
(255, 29, 213), (53, 153, 220), (246, 225, 229), (142, 82, 175)]

נניח שברצוננו להחביא את המילה Hii בתמונה.

קידוד ASCII של H הוא 72, או בבינרי : 01001000.

קידוד ASCII של i הוא 105, או בבינרי : 01101001.

לאחר שינוי הביטים הנמוכים נקבל :

[(26, 65, 164), (248, 245, 194), (174, 246, 250), (148, 95, 233),  
(188, 157, 168), (70, 167, 126), (132, 173, 97), (112, 69, 206),  
(255, 29, 213), (53, 153, 220), (246, 225, 229), (142, 82, 175)]

למשל עבור H שינינו את שלושת הפיקסלים (27, 64, 164), (248, 244, 194), (174, 246, 250) ל-

(26, 65, 164), (248, 245, 194), (174, 246, 250). ומכאן המשכנו עם i.

עליכם ליצור קובץ פייתון בשם steg\_hide.py לצורך הסתרת הטקסט בתמונה.

על הקובץ לכלול פונקציה בשם hide המקבלת כקלט תמונה (np.array), ומבצעת את התהליך

להסתרת הטקסט הנתון בתמונה. את התמונה שיצרתם שימרו בשם של התמונה המקורית

בפורמט PNG עם תוספת " \_hidden".

## חלק ב

### **המשימה**

עליכם למצוא את הטקסט שהוסתר בתוך התמונה הנתונה (hidden.png).  
בחלק זה ההסתרה נעשתה לאו דווקא בערכים הנמוכים של הביטים, ועליכם לחלץ את הטקסט החבוי בתמונה.

### **רקע**

נתונה לכם תמונה עם טקסט מוסתר. הטמנת הוקטורים הבינאריים של הטקסט בוצעה באחד משלושת המיקומים האחרונים של הוקטורים הבינאריים של הפיקסלים. עבור כל מילה, הוגרל אחד משלושת המיקומים לצורך הסתרת הטקסט.

### **דוגמה**

תמונה בגודל  $3 \times 3$  מיוצגת על-ידי המערך הבא:

[ [222, 155, 112], [222, 155, 112], [223, 156, 113],  
[223, 156, 113], [223, 152, 113], [223, 152, 113],  
[218, 146, 106], [217, 145, 108], [218, 146, 109] ]

כל ערך במערך זה מתורגם לוקטור בינארי, למשל:

155  $\rightarrow$  10011011

222  $\rightarrow$  11011110

נניח כעת כי ברצוננו להסתיר את הטקסט "hello world" בתמונה הנתונה. לצורך כך, נמיר את תווי הטקסט לוקטורים בינאריים:

h  $\rightarrow$  01101000

e  $\rightarrow$  01100101

l  $\rightarrow$  01101100

l  $\rightarrow$  01101100

o  $\rightarrow$  01101111

' ' (space)  $\rightarrow$  00100000

על מנת להסתיר את הטקסט בתמונה, הוגרל מיקום עבור כל מילה בטקסט (**התו רווח הוחשב גם כן כמילה**). ההסתרה בוצעה על-ידי החלפת הביטים בוקטורי הפיקסלים במיקום שהוגרל בביטים של וקטורי המילים. למשל, נניח כי לצורך הסתרת האות h בתמונה הנתונה, הוגרל המיקום השלישי מהסוף:

```
[ [222, 155, 112], [222, 155, 112], [223, 156, 113],  
  [..., ..., ...], [..., ..., ...], [..., ..., ...],  
  [..., ..., ...], [..., ..., ...], [..., ..., ...] ]
```

#### תמונה מקורית

```
[ [11011110, 10011011, 01110000], [11011110, 10011011, 01110000], [11011111, 10011100, 01110001],  
  [..., ..., ...], [..., ..., ...], [..., ..., ...],  
  [..., ..., ...], [..., ..., ...], [..., ..., ...] ]
```

המרת ערכי  
פיקסלים  
לוקטורים  
בינאריים

```
[ [11011010, 10011111, 01110100], [11011010, 10011111, 01110000], [11011011, 10011000, 01110001],  
  [..., ..., ...], [..., ..., ...], [..., ..., ...],  
  [..., ..., ...], [..., ..., ...], [..., ..., ...] ]
```

הטמנת הוקטור  
הבינארי של h  
במיקום  
השלישי מהסוף

```
[ [218, 159, 116], [218, 159, 112], [219, 152, 113],  
  [..., ..., ...], [..., ..., ...], [..., ..., ...],  
  [..., ..., ...], [..., ..., ...], [..., ..., ...] ]
```

#### תמונה סופית

עליכם ליצור קובץ פייתון בשם steg\_decode.py לצורך מציאת הטקסט המוסתר בתמונה.  
על הקובץ לכלול פונקציה בשם decode המקבלת כקלט תמונה (np.array), ומבצעת את התהליך  
למציאת הטקסט שהוסתר בתמונה. את הטקסט שמצאתם שמרו בקובץ בשם <id>.txt, כאשר  
במקום <id> כתבו את מספר תעודת הזהות שלכם.  
כדי לבחון את מידת נכונות הטקסט שחילצתם מהתמונה, השתמשו במילון של מילים נפוצות  
באנגלית והשוו את המילים בטקסט שמצאתם למילים במילון באמצעות מטריקת מרחק  
שתבחרו.

#### שימו לב:

- הטקסט מוצפן במיקום כלשהו בתמונה בסדר משמאל לימין ומלמעלה למטה.
- עבור כל מילה הוגרל מיקום בוקטור הפיקסלים לצורך הסתרתה. המיקום הינו אחד משלושת המיקומים האחרונים של וקטורי הפיקסלים. מיקום זה שימש להסתרת כל תווי המילה, למשל: אם עבור המילה hello הוגרל המיקום השלישי מהסוף, כל הביטים של וקטורי התווים של המילה הוסתרו במיקום זה.

- כדי להסתיר תו בודד בתמונה, דרושים 8 ערכי פיקסלים (ערך אחד עבור כל ביט בוקטור הבינארי שמייצג את התו).
- התו רווח הוסתר כמילה בפני עצמו.
- הטקסט עשוי להכיל אותיות, רווחים וסימני פיסוק ( , , ? , ! ).
- קובץ הפיתון יכול להכיל פונקציות עזר נוספות.

עבור שני הסעיפים:

עליכם לכלול בהגשה קובץ WORD המכיל הסברים להרצה, צילומי מסך של תוצאות, וכל מה שלדעתכם רלוונטי להגשה (פרט לקוד עצמו).

בהצלחה!