

אלגוריתמים בראייה ממוחשבת - 046746

רטוב 3

דניאל טייטלמן – 207734088 – Daniel.tei@campus.technion.ac.il

יאיר נחום – 034462796 – [nahum.yair@campus.technion.ac.il](mailto:nahum.yair@campus.technion.ac.il)

## פרק 1:

1. טענו את התמונות והצגנו אותן:

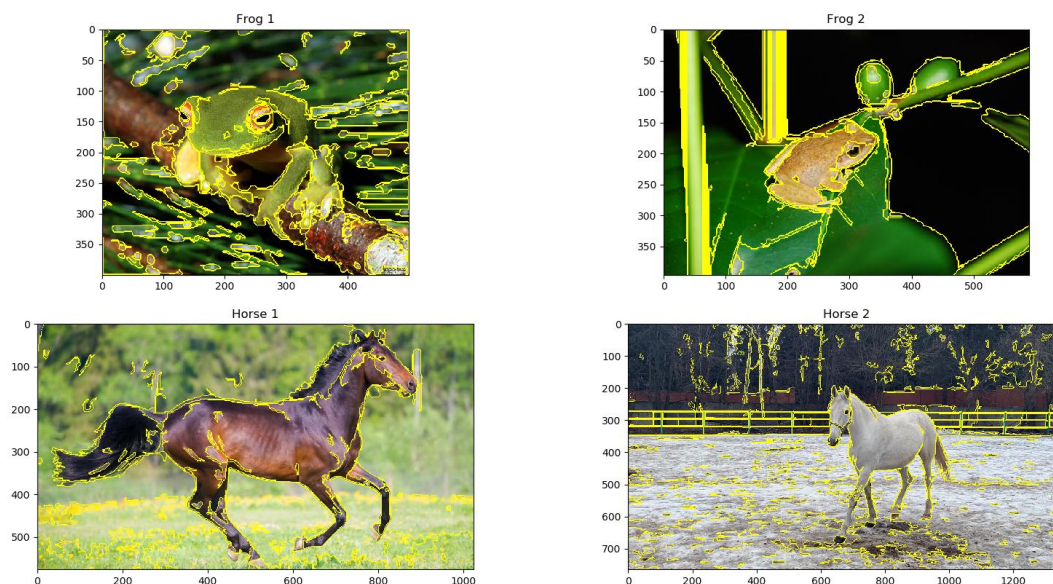


2. בחרנו את שתי השיטות הבאות לטובת פעילות הסגמנטציה:

א. שיטה קלאסית – *Felzenszwalb*.

ב. שיטה מבוססת למידה עמוקה – *DeepLabV3*.

תוצאות השיטה הקלאסית:



בעבור תמונות הצפרדעים הפרמטרים הם:

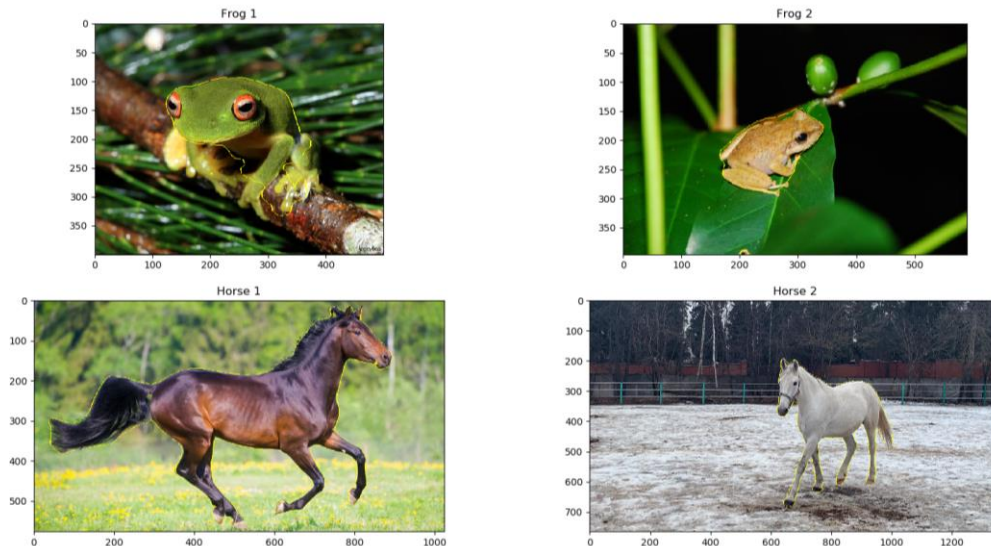
$scale=300, sigma=1, min\_size=50$

בעבור תמונות הסוסים הפרמטרים הם:

$scale=500, sigma=1, min\_size=100$

השוני בפרמטרים נובע מהעבודה כי בתמונת הסוסים יש הרבה יותר "שפות קטנות" ועל כן מקבלים אובר סגמנטציה.

תוצאות DeepLabV3:



ניתן לראות כי הסגמנטציה מצוינת, נשים לב כי ישנה שגיאה יחידה בתמונה – *Frog 1*, סה"כ תוצאות מצוינות. חסרון (או יתרון כתלות בשימוש) הוא כי מבוצעת סגמנטציה אך ורק לאובייקט המרכזי בשיטה הנ"ל לעומת השיטה הקודמת שמבצעת סגמנטציה לכלל החלקים השונים בתמונה. למרות זאת, לשיטה הראשונה יש הרבה *False positive*, לעומת השיטה מבוססת למידה עמוקה לה אין (פרט לתמונה *FROG1*) *False Positives* בכלל.

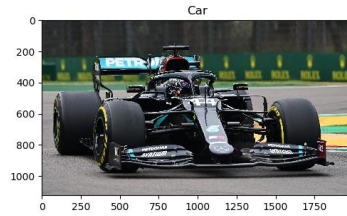
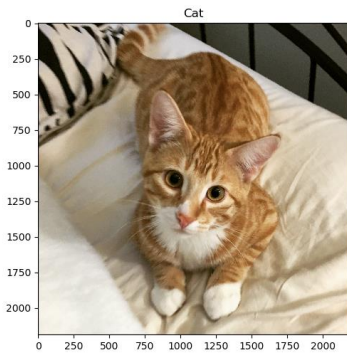
תיאור השיטות:

א. *Felzenszwalb* – שיטת סגמנטציה מבוססת המרת תמונה לגרף, אנו מחלקים את הגרף לכמה תתי סגמנטציות ראשית ולאחר מכן מבצעים צעד איטרטיבי המבוסס מרחק בין קשתות הגרף אשר על בסיסו מאחדים רכיבים (תתי סגמנטציות) ממשיכים בתהליך עד להתכנסות. ב. *DeepLabV3* – מודל המבוסס על מודל *DeepLabV2* (כאשר הוסר מודל *Conditional random field*), במודל זה השתמשו בקונבולוציות מסוג *atrous* אשר מאפשרות סגמנטציה בסקאלות שונות וכך להתייחס לפיצרים משמעותיים בסקאלות שונות. בנוסף משתמשים בדומה לגרסה *2V* בפירמידה של אוסף הסקאלות השונות של הפיצרים, אך מבוצעת קונבולוציה נוספת ו – *Batch norm* נוסף.

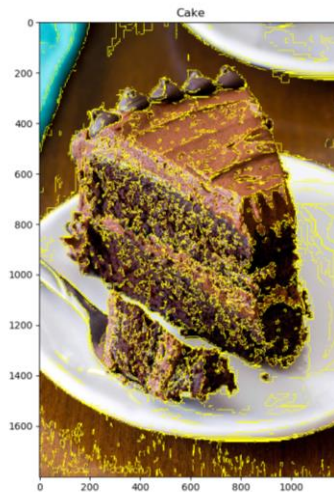
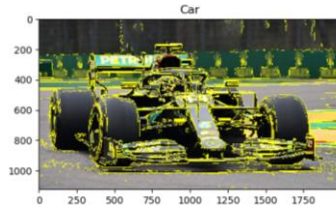
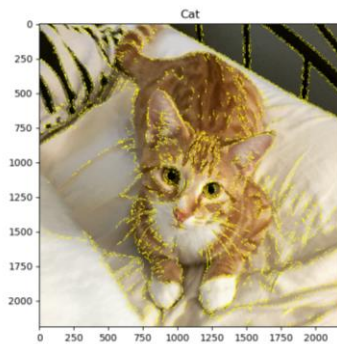
לסיכום, ניתן לראות כי במקרה זה השיטה המבוססת למידה עמוקה טובה משמעותית מן השיטה הקלאסית. אך לשיטה הקלאסית יתרון משמעותי בזיהוי שפות נוספות.



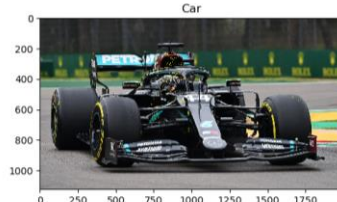
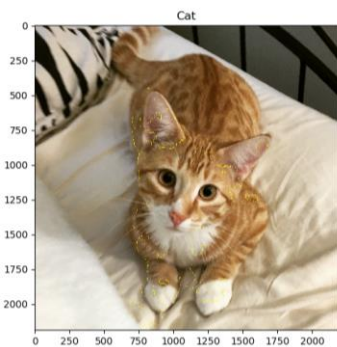
3. התמונות שבחרנו:  
חתול, מכונית ועוגה.



4. תוצאות שיטות הסגמנטציה:  
שיטה קלאסית:



שיטה מבוססת למידה עמוקה:



- כפי שנתין לראות במקרה זה השיטה הקלאסית מניבה תוצאה טובה יותר משמעותית על פני כל שלושת התמונות, כאשר נשים לב כי בעבור תמונת החתול תוצאת הרשת היא הטובה ביותר מבין שלושת הניסיונות אך עדיין פחותה מן השיטה הקלאסית. נשים לב כי בעבור תמונת העוגה ישנו רעש משמעותי בעבור שתי השיטות ואין מצליחות לבצע סגמנטציה בצורה טובה. התוצאה הקיצונית ביותר היא הרכב כאשר השיטה הקלאסית מצליחה מצוין ואילו השיטה הלומדת נכשלת לחלוטין ומצליחה מעט לבצע סגמנטציה לשפת הנהג. לפי דעתנו השיטה הקלאסית טובה יותר במקרה זה מפני שהתמונות רועשות מאוד (במובן שפות ולא רעש בתמונה), מצולמות באוריינטציות שונות ובנוסף לא בהכרח נמצאות כחלק מסט האימון בעבור הסגמנטציה ועל קשות לסגמנטציה ברשת.
5. על מנת לשפר את תהליך הסגמנטציה היינו נציע כמה אפשרויות:
- א. ביצוע החלקה לתמונה לפני הפעלת הסגמנטציה ע"י פילטר גאומי לדוגמא.
  - ב. ביצוע אימון בעבור השיטה מבוססת למידה עמוקה עם אוגמנטציות רבות הכוללות, שינוי אוריינטציה או רעש.
  - ג. ביצוע אימון על *Dataset* גדול יותר.
  - ד. ליצור *Ensemble* של שיטות שונות לביצוע סגמנטציה ולקחת את האיחוד המייצג ביותר של כלל התוצאות של השיטות השונות.
6. טענו את מודל 16VGG בדומה לש"ב הקודמים במצב *Pretrained*.
7. טענו את תמונת הפרה שסופקה וביצענו *Feedforward* עם הרשת:

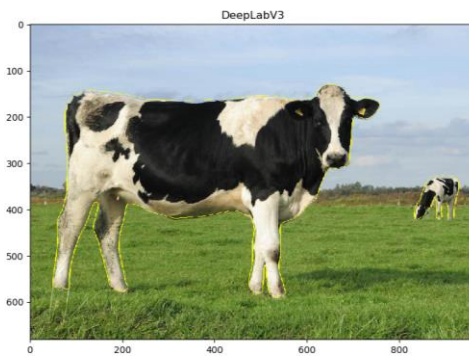
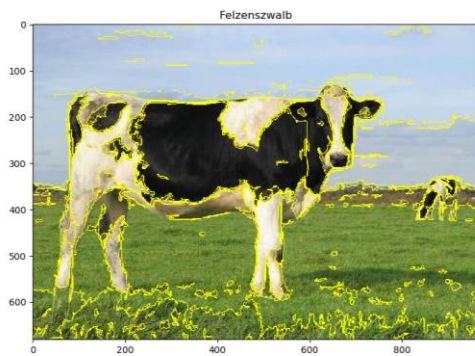


אם נבדוק באינטרנט מהו *OX* נגלה שאכן הכל בסדר:





8. ביצוע סגמנטציה לתמונה:



נשים לב כי הסגמנטציה באמצעות הרשת טובה משמעותית ופחות רגישה לרעשים שונים, אך מתמקדת אך ורק בפרות. מכאן נמשיך הלאה רק עם שיטת הלמידה עמוקה.

9. ראשית כל חילצנו את תמונת הפרה מן הרקע:



כעת נרצה לשים את הפרה על הרקע הבא:



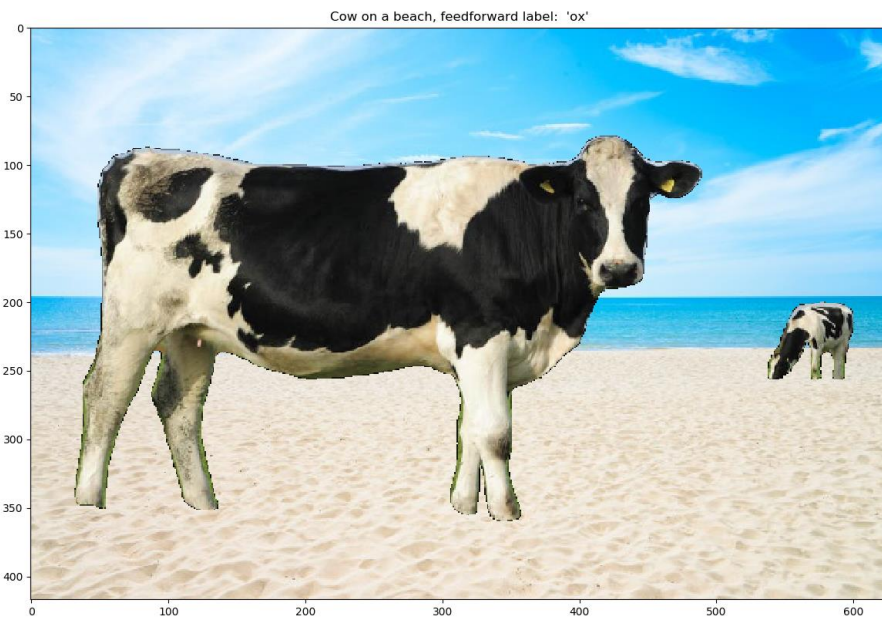
נאחד את הרקע והתמונה החתוכה של הפרות, אך ראשית נעשה *Rescale* לתמונת הפרות לגודל תמונת הרקע.



כמו שאמרנו לפני כן כעת נאחד.



ניתן לראות כי התוצאה טובה, אך השפות משמעותיות מאוד, בשל העובדה כי התמונות שונות מאוד זו מזו. 10. לבסוף נכניס את התמונה שיצרנו ל 16VGG ונראה כעת מה תהיה הפרדיקציה.



קיבלנו שוב סיווג נכון בדומה לסעיף 7.



## פרק 2:

1. הקלטנו סרט והמרנו את קובץ הוידאו לתמונות. שתי תמונות מתוך הסרט:



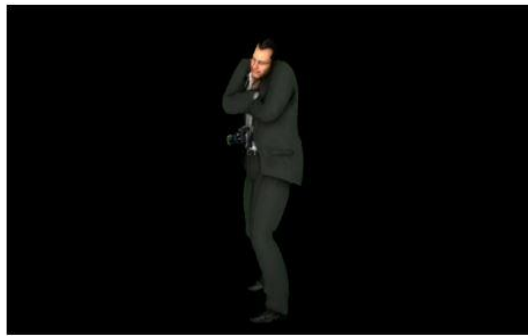
2. אותן התמונות לאחר סגמנטציה בעזרת רשת הנוירונים מפרק 1  
וחיתוך בהתאם למסיכה שקיבלנו:



3. הפעלנו סגמנטציה קלאסית ע"פ לינק שניתן בתרגול לביצוע סגמנטציה פשוטה לפי צבע:

[/https://realpython.com/python-opencv-color-spaces](https://realpython.com/python-opencv-color-spaces)

המרנו ל HSV וביצענו סגמנטציה לפי צבע שכן צבע הרקע הירוק הבהיר אחיד וקל לסגמנטציה שכזו. היינו צריכים להבין מהם ערכי HSV של הרקע ולפלט (בדומה ל *threshold segmentation* על תמונות שחור לבן) בהתאם:





4. הוספנו יד על תמונת הרקע לאפקט כאילו האיש הרוקד רוקד עליה.  
הרקע ליד היה לבן כך ששוב ביצענו סגמנטציה קלאסית לפי צבע  
הרקע.

הוספנו לרקע ע"י מסיכה (גם מסיכה הופכית בכדי לקחת את תמונת  
הרקע המקורית), *resize* של התמונה וחיבור ב *offset* על גבי הרקע.  
הוספנו בצורה דומה את האיש הרוקד ואת הסרט שיצרנו בעצמנו  
(ארוך יותר ולכן האיש הרוקד חוזר לרקוד מההתחלה בשלב מסוים).

לינק לסרט ששותף ב *youtube* :

[https://youtu.be/uwf\\_uDdlcEw](https://youtu.be/uwf_uDdlcEw)

שתיים מהתמונות בסרט:

