עיבוד תמונה – תרגיל 1

הראל יונה, 315261727

# מבוא

סרטון וידאו דיגיטלי מורכב מרצף של תמונות (פריימים), וכל תמונה מורכבת מפיקסלים אשר מכילים את שלושת ערכי RGB(במקרה של סרטון שחור-לבן כל פיקסל מיוצג על ידי ערך יחיד).  
מטרת התרגיל היא לזהות באופן אוטומטי 'קפיצת סצנה' בקטע וידאו נתון. קפיצת סצנה היא מעבר מידי וחד בין שתי סצנות שונות.  
הרעיון המרכזי שבו השתמשנו לזיהוי המעבר הוא ניתוח סטטיסטי של התפלגות עוצמות הפיקסלים בין פריימים עוקבים. ההנחה היא שבתוך סצנה רציפה, ההתפלגות הסטטיסטית של הפיקסלים תהיה דומה מאוד מפריים לפריים. לעומת זאת, בנקודת המעבר החדה, התפלגות זו תשתנה באופן דרמטי.

על מנת לפתור את הבעיה, המרתי את הסרטון לגווני שחור-לבן והשתמשתי היסטוגרמה מצטברת לערכי הפיקסלים.

היסטוגרמה מצטברת זו מייצגת עבור כל גוון אפור (0-255) כמה פיקסלים קיימים בתמונה אשר ערכם שווה או קטן לאותו גוון אפור.  
  
בתרגיל התמודדנו עם שתי קטגוריות של סרטוני וידאו:

1. קטגוריה 1**:** סרטונים "נקיים" שבהם הפריימים בתוך כל סצנה כמעט זהים, והשינוי המשמעותי היחיד הוא קפיצת הסצנה עצמה.
2. קטגוריה 2**:** סרטונים מורכבים יותר, אשר בנוסף לקפיצת הסצנה כוללים גם שינויים ויזואליים אחרים, כגון שינוי בחדותשל התמונה.

# אלגוריתם

הטכניקה שהשתמשתי בה לפתרון הבעיה היא כדלהלן:

1. המרת הסרטון לגווני אפור בעזרת הנוסחה -
2. חישוב היסטוגרמה המצטברתעבור כל פריים.
3. חישוב סכום ההפרשים בין כל ערך של גוון אפור של ההיסטוגרמות המצטברות של כל זוג פריימים עוקבים.
4. מציאת הפריימים שעבורם סכום ההפרשים מקסימלי

האלגוריתם הנ"ל מתאים עבור סרטונים משתי הקטגוריות

# פרטי מימוש

האלגוריתם יושם בשפת Python תוך שימוש במספר ספריות מרכזיות:

קריאת נתונים: בשלב הראשון, קובץ הווידאו הומר למערך נתונים (tensor) המייצג את רצף הפריימים. המרה זו התבצעה באמצעות הפונקציה mp.read\_video מספריית mediapy.

עיבוד וניתוח: כל הפעולות המתמטיות על המערך, כולל המרה לגווני אפור, חישוב היסטוגרמות מצטברות, וחישוב ההפרשים בין פריימים, בוצעו באמצעות ספריית .numpy השימוש בספרייה זו מאפשר ביצוע יעיל של פעולות וקטוריות על מערכי פיקסלים גדולים.

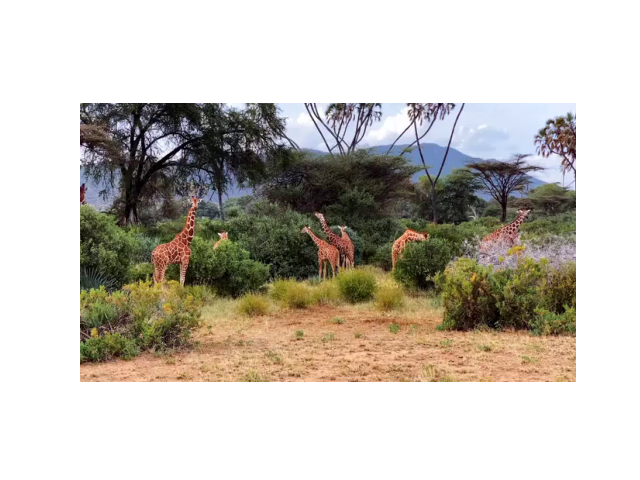
ויזואליזציה ותיקוף: לצורך וידוא התוצאות, נכתבה פונקציית עזר (show\_frame) המציגה פריים נתון. הצגת התמונות התבצעה באמצעות ספריית matplotlib.

# תוצאות קטגוריה 1

בסרטון החיתוך התבצע בין הפריימים 99 , 100:

בסרטון החיתוך התבצע בין הפריימים 149, 150:

 תמונה שמכילה יונק, בחוץ, חיות בר, צבי

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

# תוצאות קטגוריה 2

כפי שכבר נאמר – ניתן לראות שבמהלך הסירטון יש שינוי בחדות התמונות, אך הדבר לא השפיע על נכונות האלגוריתם והתוכנית הצליחה למצוא את מיקום החיתוך בהצלחה

בסרטון הראשון החיתוך התבצע בין הפריימים 174, 175:

 תמונה שמכילה עש ופרפרים, חסרי חוליות, חרק, מאבק

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

בסרטון בשני החיתוך התבצע בין הפריימים 74, 75:

 תמונה שמכילה יונק, סנאי, בחוץ, מכרסם

תוכן בינה מלאכותית גנרטיבית עשוי להיות שגוי.

# מסקנות

בתרגיל זה יישמנו בהצלחה אלגוריתם לזיהוי קפיצות סצנה חדות בקטעי וידאו. האלגוריתם התבסס על חישוב סכום ההפרשים המוחלטים בין היסטוגרמות מצטברות.  
מהתוצאות שהוצגו, ניתן להסיק כי השיטה יעילה ואמינה, האלגוריתם הצליח לזהות נכונה את נקודות החיתוך בכל סרטוני המבחן, הן בסרטונים ה"נקיים" מקטגוריה 1 והן בסרטונים המאתגרים מקטגוריה 2.