

Image Processing - Exercise 1

Neriya Ben David, neriyabd, 206698581

מבוא

1. המטרה העיקרית במשימה היא לאתר את הפריים בו מתרחש מעבר בין סצנות שונות בסרטון נתון. כדי להשיג את המטרה, יש להבין איך להשתמש בשינויים בין דרגות האפור בפיקסלים השונים בכל פריים ופריים. הרעיון המרכזי מההרצאה שבחרתי להשתמש בו הוא לבחון את ההיסטוגרמה של כל פריים, ובפרט את ההיסטוגרמה המצטברת. השתמשתי בטכניקה של השוואת ההיסטוגרמה מצטברת בין שתי תמונות (פריימים). הסתכלות על כל ההיסטוגרמה כוקטור חד ממדי שמכיל את כל דרגות האפור של הפיקסלים לפי סדר, ומדידת המרחק באמצעות פונ' מרחק.

2. ההבדלים בין שתי הקטגוריות נובעות מכך שבקטגוריה הראשונה, שתי הסצנות מאוד הומוגניות בדרגות האפור שלהן ואין בהם כל כך שינויים ואילו, בקטגוריה השנייה ישנם סצנות בהם הרזולוציה משתנה, בתהליך קוונטיזציה. שינוי זה משפיע על דרגת האפור בין שני הפריימים שביניהם מתרחשת ירידת הרזולוציה וכתוצאה מכך ישנו קושי להבחין מתי ישנו שינוי רזולוציה ומתי ישנה החלפת סצנה. כלומר, עלינו לזהות באלגוריתם שהצענו מתי תתרחש החלפת סצנה ולא לטעות עבור ירידת איכות ברזולוציה.

אלגוריתם

1. שלבי האלגוריתם שמימשתי הינם :
 - הכנת ההיסטוגרמה עבור כל פריים בסרטון.
 - חישוב ההיסטוגרמה המצטברת עבור כל פריים.
 - שימוש בפונ' מרחק בין כל שתי ההיסטוגרמות צמודות על מנת לחשב את ההפרש בין דרגות האפור בין כל שני פריימים סמוכים.
 - החזרת אינדקס הקואורדינטה המקסימלית בוקטור שהתקבל מפונ' המרחק והקואורדינטה שאחריה. (בשלב זה האלגוריתם מחזיר את הפריימים הסמוכים שבהם האלגוריתם זיהה שהתרחשה מעבר הסצנה).

2. לא ביצעתי שינויים באלגוריתם בין הסצנה הראשונה לשנייה.

פרטי מימוש

א. ראשית, טענתי את הסרטון למערך תלת – ממדי באמצעות שימוש בפונ' `read_video` מהספרייה `mediapy`, כך שכל מטריצה דו – ממדית בתור המערך מייצגת פריים מהסרטון בדרגות אפור. בכדי לעבוד עם הסטוגרמות נרמלתי את ערכי האפור ל $[0, 1]$ והעברתי באמצעות פונ' `reshape` את המערך המייצג את הסרטון לדו ממדי, כך שכל פריים ייוצג במערך חד מימדי, כלומר כל שורה במטריצה מייצגת הסטוגרמה של פריים.

- חישבתי את ההיסטוגרמה המצטברת באמצעות פונ' `cumsum` על המערך הדו ממדי.
- ראשית, חישבתי את ההפרש של פריימים עוקבים באמצעות פונ' `diff`. לאחר מכן על המערך שהתקבל השתמשתי בפונ' המרחק נורמה 1 ונורמה 2, כדי לקבל מספר ממשי עבור מרחק בין כל שני פריימים כדי שאוכל להשוות בין כל שני פריימים סמוכים, עד כמה הם "רחוקים".
- ממימוש האלגוריתם אינדקסים אלו אינדקסים אילו הינן האינדקסים שהאלגוריתם מזהה בתור הפריים האחרון בסצנה הראשונה והפריים הראשון בסצנה השנייה בהתאמה, כיוון שבהם התקבלו המרחקים הכי גדולים בין שני פריימים סמוכים, המתארים שינוי סצנה.
- השתמשתי בספרייה `numpy` על מנת לבצע מניפולציות על נתוני הסרטון. השתמשתי בה כדי "לשטח" כל פריים, לחשב את ההסטוגרמה הצוברת ולחשב מרחקים בין כל שני פריימים סמוכים (מרחק ואז הפעלת נורמה על המרחק). הסיבה לכך שהשתמשתי בה היא כיוון שספרייה זו הינה יעילה בזמן הריצה שלה ביחס למימוש עצמי של פונ' אלו.

- עשיתי שימוש ב-`matplotlib` בכדי לייצר את הגרפים עם הנתונים שרציתי להציג.
- השתמשתי ב-`pandas` על מנת ליצור גרפים דו מימדיים על מערך חד מימדי, השימוש ב-`DataFrame` אפשר לי לייחס את ציר ה-x בגרף למספר הפריים בצורה קלה ונוחה.

ב. עשיתי שימוש של היפר-פרמטרים עבור כמות התאים בהיסטוגרמה. בחרתי שיהיו 256 תאים על מנת לתת תא לכל דרגת אפור שיש בפריים. בכך שהשתמשתי בייצוג של דרגות אפור בערכים $[0, 255]$ יכולתי לתת לכל דרגת אפור תא. בנוסף, בחרתי $\text{range} = (0, 256)$ על מנת שבכל תא יהיה אפשרות לייצוג 256 דרגות אפור. כמו כן, לא ראיתי צורך להשתמש בהיפר-פרמטרים נוספים שכן האלגוריתם בו השתמשתי לא נדרש אליהם. חישוב הסטוגרמה מצטברת אינו דורש שימוש בהיפר – פרמטרים.

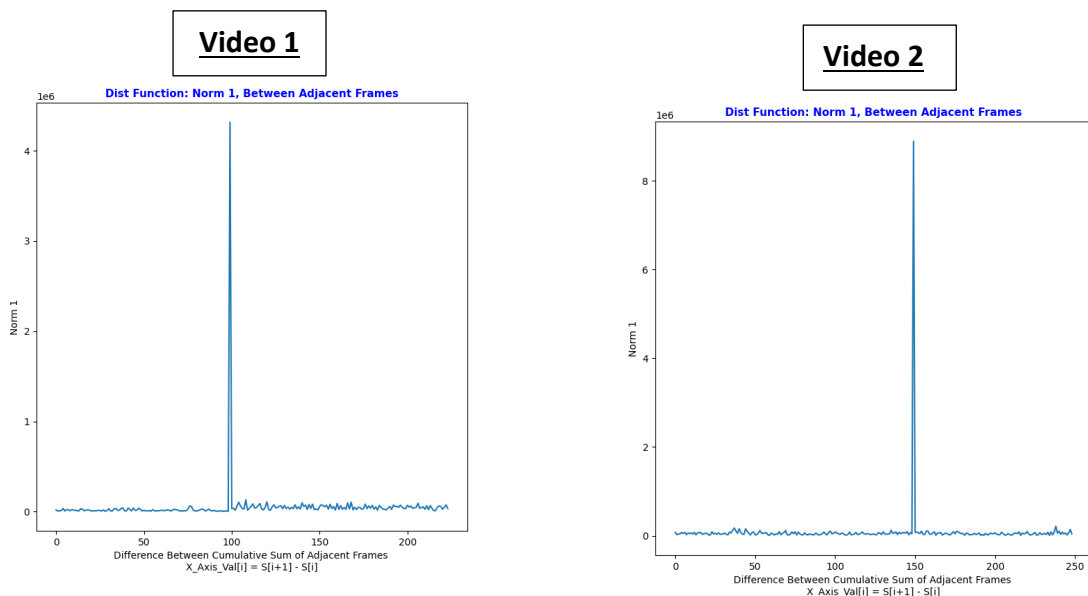
ג. הקשיים שבהם נתקלתי הם עבודה עם ספריות שלא היה לי ניסיון בהם, דוג' `mediapy`, `matplotlib`. עם קשיים אלו התמודדתי באמצעות תרגול ה-`bootcamp` ותרגול עצמי עם שלל פונ' מהספריות הנ"ל. היה לי קושי לחשוב על דרך בה אוכל לממש את האלגוריתם בצורה יעילה. התמודדתי עם קושי זה באמצעות שימוש בספריית `numpy` בה ניתן לבצע חישובים מהירים על מערכים (דוג': חישוב סכום מצטבר, יצירת ההסטוגרמה, חישוב מרחק בין פריימים סמוכים, הפיכת מטריצה לוקטור ועוד...)

תוצאות קטגוריה 1

א. תוצאות הפריימים הסמוכים שבהם התבצעה מעבר סצנה :

The Results of Cut Detection of Category 1 are:
video 1: (99, 100), video 2: (149, 150)

ב. נתבונן בגרף המחשב את הפרשי ההסטוגרמות הצוברות בין כל שני פריימים בקטגוריה 1.



בשני המקרים ניתן לראות שישנו מרחק גדול בין הפרשי ההיסטוגרמות במקום אחד בלבד ולכן נסיק כי זהו מעבר הסצנה. (בין 99 – 100 בסרטון הראשון, בין 149 – 150 בסרטון השני)

תוצאות קטגוריה 2

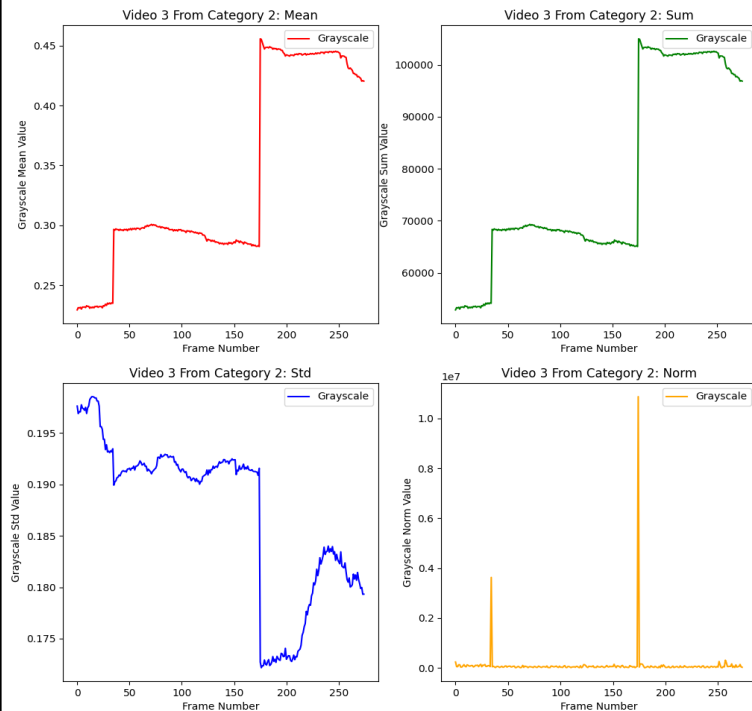
א. תוצאות הפריימים הסמוכים שבהם התבצעה מעבר סצנה :

The Result of Cut Detection of Category 2 are:
video 3: (174, 175), video 4 (74, 75)

ב. נתבונן בגרף המראה מגוון תכונות לאורך הפריימים השונים (הפרש הסטוגרמות, סכימה, ממוצע וסטיית תקן בכל פריים). השיפוע בגרף מעיד על ההבדל בתכונות בין כל שני פריימים שכנים.

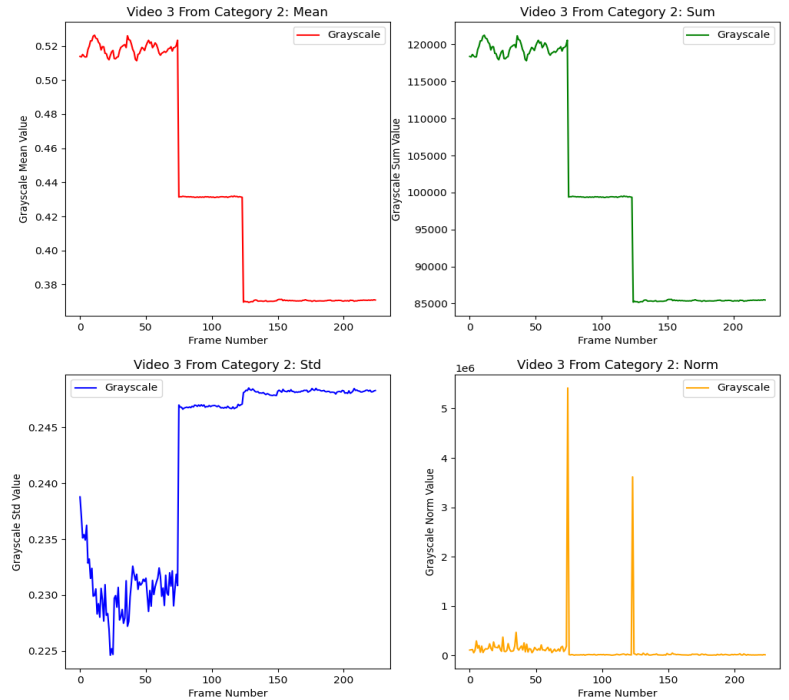
Video 3

Frames Features, Grayscale Values: [0, 1]



Video 4

Frames Features, Grayscale Values: [0, 1]



ניתן לראות כי ישנם 2 זוגות פריימים סמוכים בהם הפרש ההסטוגרמה הצוברת (מימין למטה בשני הגרפים) הינו גדול, במקרה אחד זה שינוי ברזולוציה ואילו במקרה השני זו חילוף סצנה. ההשערה הראשונית היא שחילוף סצנה יהווה הפרש גדול יותר בין דרגות אפור שונות, ואילו ניתן לראות מהממוצע, סכום ובמיוחד מהסטיית תקן את השינוי בין פריימים סמוכים, כלומר מבססת את ההנחה שחילוף הסצנה התבצע כאשר הפרש ההסטוגרמה המצטברת היה מירבי.

מסקנות

בתרגיל זה ראינו את השימוש בהיסטוגרמות, ובהיסטוגרמות צוברות. במהלך התרגיל הבנתי את החשיבות בהסתכלות על ההסטוגרמה צוברת המאפשרת להביט על ההסטוגרמה גם ממימד ה"מרחק" בין הפיקסלים השונים בתמונה. ממצא חשוב שהפקתי הוא שאלגוריתם המשתמש בהסטוגרמה צוברת יביא לביצועים טובים יותר כאשר השינוי בין שתי סצנות הוא מהותי, כלומר סביבה ורקע המשנים באופן מהותי את ערכי דרגות האפור עבור מיקומים זהים בין הפריימים. בנוסף, הבנתי כי מציאת חתך בין סצינה יכול להיות "מורעש" ע"י גורמים נוספים, דוג' קוונטיזציה ועוד, ואילו כל אלגוריתם המוצא מעבר סצנה מחוייב לקחת פקטורים אלו בחשבון ולסווג מתי ישנו מעבר סצנה או מניפולציה אחרת שאינה קשורה למעבר הסצנה.