

Image Processing - Exercise 1

Agam Hershko, id_214193831, 214193831

מבוא

מטרת התרגיל היא להבחין בין מעבר סצינות במהלך סרטון ("קאטים"). הכלי המרכזי מהחומר אותו ראינו בכיתה שאותו ניישם הוא היסטוגרמות ופעולות שניתן להפעיל עליהן. נעסוק בהיסטוגרמות של תמונות מסוג grayscale שקיבלנו מפריימים של סרטונים, ניצור היסטוגרמות צוברות ונפעיל עליהן פעולות מתאימות כדי להכריע מהם שני הפריימים ביניהם קורה מעבר הסצנה.

הסרטונים בתרגיל מחולקים לשתי קטגוריות- קטגוריה 1 וקטגוריה 2. בקטגוריה 1, מתרחשים שינויים קטנים בסרטון אך התנאים הכלליים בו נשמרים- בעקבות כך גם לא מתבצעים שינויים משמעותיים בהיסטוגרמות למעט במעבר בין הסצנות. בקטגוריה 2, מתרחשים בלפחות סצנה אחת מהשתיים בסרטון שינויים בטשטוש ובתאורה (אני סבור שזה קשור לקוונטיזציה) כך שרמת האור בתמונה משתנה משמעותית, מה שמוביל לשינוי משמעותי בהיסטוגרמה. במצב זה ההבדל המשמעותי ביותר בין הפריימים עדיין קורה במהלך מעבר הסצנה ולכן הצלחתי להשתמש באותו האלגוריתם עבור שתי הקטגוריות.

אלגוריתם

מטרת האלגוריתם היא לקלוט מידע מהפריימים של הסרטון ולהשוות אותם לפריימים אחרים כדי לזהות את מעבר הסצנה על ידי שימוש בהיסטוגרמות וכיוב'.

נעבור על כל הפריימים בסרטון בלולאה, כאשר כל פריים מהסרטון מייצג תמונה שאותה נייצג באמצעות מערך דו מימדי של פיקסלים. התמונה למעשה מתקבלת כמערך תלת מימדי כאשר לכל פיקסל יש 3 ערוצים צבעוניים (פורמט RGB), אך נמיר את התמונה לgrayscale כאשר לכל פיקסל יהיה ערך שמייצג את עוצמתו בין 0 ל255 (כאשר 0 זה שחור ו255 זה לבן).

ניצור מהתמונה בגווי ה grayscale היסטוגרמה עבור התמונה (שלב ההמרה לgrayscale הכרחי- אחרת היינו צריכים ליצור היסטוגרמה בנפרד עבור כל ערוץ צבע).
כעת נרצה להשוות בין ההיסטוגרמות על מנת לזהות את השינוי בסצנה (כל היסטוגרמה מייצגת פריים)- למדנו בכיתה שהדרך הנכונה לעשות זאת היא על ידי מרחק בין היסטוגרמות מצטברות.
לכן עבור כל היסטוגרמה, ניצור את ההיסטוגרמה המצטברת שלה ונחשב את הסכום שלה. על מנת לחשב את המרחק בין שתי היסטוגרמות מצטברות, נחשב את הערך המוחלט של ההפרש בין הסכום של ההיסטוגרמה המצטברת עבור הפריים הנוכחי לסכום של ההיסטוגרמה המצטברת עבור הפריים הקודם.
האלגוריתם מוצא את הפריימים עבורם מתקבל ההפרש המקסימלי.

נשים לב שהאלגוריתם יעבוד רק בתמונה עם קאט מוגדר אחד ועבור תמונה מרובת קאטים נצטרך לנקוט בגישה אחרת. האלגוריתם יעבוד עבור 2 הקטגוריות כיוון שהוא מוצא את המרחק המקסימלי בין 2 היסטוגרמות מצטברות רצופות, ייתכן ואם היו שינויים קיצוניים ביותר בתאורה במהלך הסצנה היה יכול להתקבל פלט שגוי של האלגוריתם לפי האלגוריתם שבחרתי לממש.

פרטי מימוש

הפונקציה המרכזית בקוד (ולמעשה היחידה) בקוד שלנו היא main, שמקבלת נתיב לקובץ של סרטון וסוג הסרטון (האם הסרטון שייך לקטגוריה 1 או 2- כיוון שהאלגוריתם מתאים ל-2 הקטגוריות לא נשתמש בפרמטר זה) ומחזירה tuple של שני אינדקסים- הפריים שלפני ואחרי מעבר הסצנה.

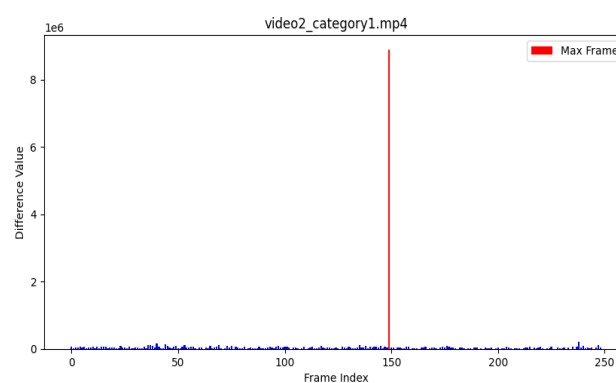
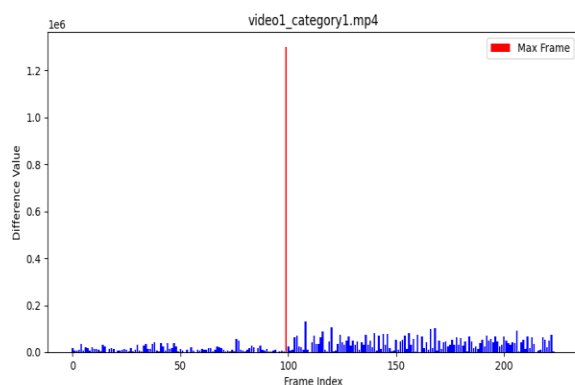
נשתמש בפונקציה read_video של mediapy שמקבלת את נתיב הסרטון על מנת לשמור למערך (תלת מימדי) את הפיקסלים של התמונה עבור פריימים בסרטון (לאחר מכן נמיר לnp.array). כמו כן, נאתחל רשימה עבור מרחקים בין היסטוגרמות רצופות (hist_diff) ומשתנה עבור הסכום של ההיסטוגרמה המצטברת בפריים הקודם (prev_sum), שישמשו אותנו בהמשך.

נעבור בלולאה על הפריימים בסרטון ונמיר את התמונה (כל פריים מייצג תמונה) לgrayscale על ידי שימוש בפונקציה convert עם הערך המתאים של "L". ניצור היסטוגרמה על ידי שימוש בפקודה np.histogram שמקבלת את התמונה, את מספר העמודות (נגדיר אותו כ-256) ואת הטווח (שנגדיר בין 0 ל-255). נחשב את ההיסטוגרמה המצטברת על ידי שימוש בפקודה np.cumsum ונחשב את הסכום שלה על ידי שימוש בפקודה np.sum. עבור כל פריים מלבד הראשון, נחשב את הערך המוחלט (באמצעות np.abs) של ההפרש בין הסכומים של ההיסטוגרמות המצטברות (המיוצגות על ידי המשתנים prev_sum, current_sum) ונוסיף אותו ל hist_diff. לאחר מכן, נעדכן את הסכום של ההיסטוגרמה המצטברת בפריים הקודם בערך של הפריים הנוכחי.

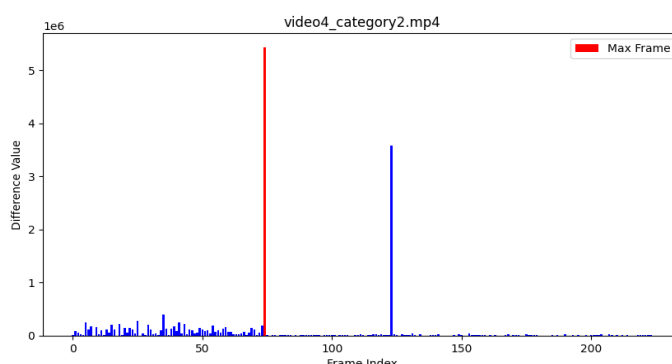
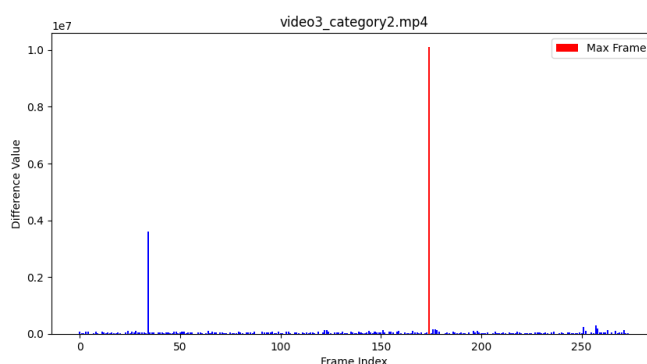
נשתמש בnp.argmax על מנת למצוא את האינדקס של האיבר המקסימלי ברשימת המרחקים ולהחזיר tuple של האינדקס של פריים זה ואת האינדקס של הפריים הבא.

תוצאות קטגוריה 1

להלן אציג גרפים שמתארים את המרחק בין ההיסטוגרמה בפריים הנוכחי לפריים הקודם לו. למעשה מדובר בייצוג ויזואלי של מערך ההפרשים hist_diff, כאשר העמודה הבולטת היא של האינדקס שבו מתקבל ההפרש המקסימלי. ניתן להבחין שמדובר בגרפים מהקטגוריה הראשונה כיוון שהם פשוטים ומכילים רק פיק משמעותי אחד.



תוצאות קטגוריה 2



נשים לב שבקטגוריה השנייה יש כבר הבדל גדול בצורת הגרף לעומת הקטגוריה הקודמת- ניתן להבחין בעמודה נוספת שנובעת משינוי קוונטיזציה בתמונה במהלך אחת הסצנות (בוידאו 3- גרף שמאל מדובר בסצנה הראשונה ובוידאו 4- גרף ימין מדובר בסצנה השנייה). האלגוריתם של קטגוריה 1 יעבוד עבור קטגוריה 2 כיוון שמדובר במרחק המקסימלי בין היסטוגרמות שגדול מהמרחק בין היסטוגרמות שנובע משינוי קוונטיזציה (הבדל בתאורה ובטשטוש בסצנה). הסיבה שזה מתאפשר היא השימוש בהיסטוגרמה צוברת, שרגישותה לשינויי גוון לא גבוהה.

מסקנות

1. קריטי להשתמש בהיסטוגרמות בכלל ובהיסטוגרמת צוברת בפרט על מנת לקבל מידע על פריימים שונים בסרטון ועל ההבדלים ביניהם. לשם כך, השתמשנו בהגדרה המרחק בין היסטוגרמת שלמדנו בכיתה שמיישמת הפרש בין סכום היסטוגרמות צוברות (אם היינו מסתפקים בהיסטוגרמות רגילות לא היינו יכולים להבחין בין גוונים דומים ביעילות).
2. מניפולציות על תמונות יכול לעזור להסיק עליהן מסקנות, אך גם יכול לעוות אותן בצורה בלתי הפיכה (בכפוף לניהול זיכרון חכם) ולכן צריך להשתמש בכלים שברשותנו בתבונה רבה
3. הפתרון שלי לתרגיל עובד תחת מגבלות מסוימות שיש לתת עליהן את הדעת- קיים קאט יחיד שמהווה את המרחק המקסימלי בין היסטוגרמות ואין שינוי במהלך הסצנה שנובע מקוונטיזציה או

מסיבה אחרת שגורם לקח שיוצר מרחק גדול יותר מהמרחק בעת מעבר הסצנה. כמו כן, לא התייחסנו כלל למיקום הפיקסלים בתמונה (לפי הגדרת ההיסטוגרמה) ולכן יכול להיות שפספסנו מעבר סצנה ששמר על אותה ההיסטוגרמה בקירוב אבל שינה לחלוטין את התמונה.

4. בעקבות כך, ייתכן ונצטרך להפעיל במצבים שונים על התמונות פעולות נוספות שראינו כמו שיווי ההיסטוגרמה, תיקון גמא וכו'.