The Iby and Aladar Fleischman
Faculty of Engineering
Tel Aviv University

הפקולטה להנדסה
ע"ש איבי ואלדר פליישמן
אוניברסיטת תל אביב

שיפור אלגוריתם לזיהוי אדם ע"י צורת הליכה

פרויקט מס' 22-1-1-2565

דו"ח סיכום

מבצעים:

יונתן רוסין 324035369

אופיר סולומון 322728809

מנחה:

מר חן כהן אוניברסיטת ת"א

מקום ביצוע הפרויקט:

מעבדת אופטיקה של פרופ' דוד מנדלוביץ'

תוכן עניינים

רשימת איורים

תקציר

הפרויקט שלנו עוסק ב-gait recognition, תחום הכולל שימוש בטכניקות ראייה ממוחשבת לחילוץ וניתוח מאפיינים מרחביים-זמניים שונים ע"פ הליכתו של אדם, ומאפשר לבצע בין היתר זיהוי אנשים ע"פ אותם מאפיינים, ניתוח בריאותי וכו'. כלומר, לתחום זה יש חשיבות למטרות ביטחוניות, שירותי בריאות, חקירות משפטיות ועוד.

מטרת הפרויקט היא להוכיח כי ניתן לבצע קלסיפיקציה בין אנשים, אך ורק על פי צורת ההליכה שלהם והמבנה הפיזי שלהם, המתואר, במודל שבו השתמשנו, על פי 17 נקודות המתארות את גופו של האדם.

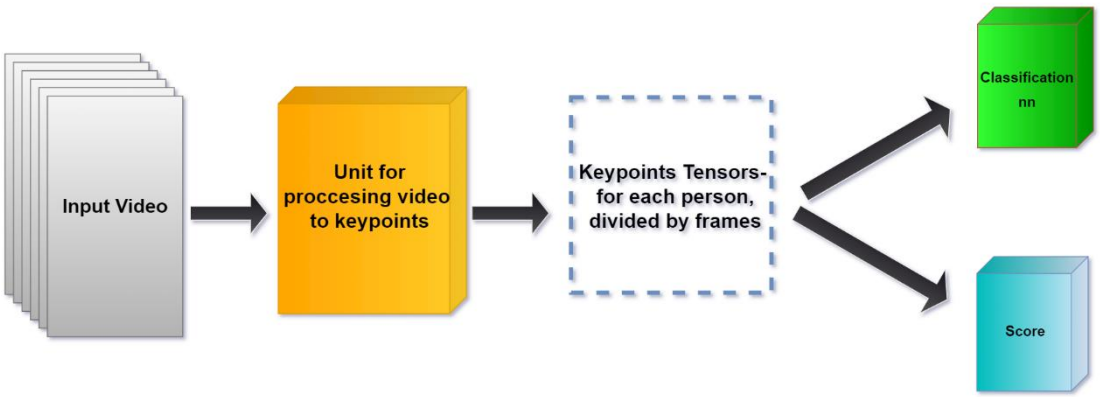
בפרויקט שלנו השתמשנו במודל pose estimation הנקרא MMPOSE [1], שממפה בצורה תלת ממדית את גופו של אדם באמצעות 17 נק' מפתח (skeleton). מצאנו דרך לשפר את הצלחת המודל גם בתנאי צילום משתנים בהם הרזולוציה של אדם בסרטון קטנה מדי מה שמוביל לכישלון במודל המקורי, והוכחנו שלאחר שיפור הצלחת המודל משימת הקלסיפיקציה מצליחה באחוזי הצלחה גבוהים- 95% הצלחה עבור הדאטה שצילמנו .

אופן שיפור מודל ה-MMPOSE מתבצע, על ידי מודל object detection הנקרא YOLOv7 [2] שבעזרתו אנחנו מפרקים כל פריים לאנשים שבסרטון, לאחר מכן העלאת הרזולוציה באמצעות מודל super resolution הנקרא Real-ESRGAN [3], ורק לאחר מכן מציאת skeleton (והטנזורים המכילים את נקודות המפתח) בעזרת MMPOSE.

תוצרי הפרויקט הם:

1. קוד python שמקבל סרטונים של אנשים הולכים, ומחזיר טנזורים של נקודות המפתח, ו-GUI למיקום הנקודות על הגוף של כל אדם.
2. רשת קלסיפיקציה מבוססת LSTM, שאומן על dataset המכיל שני אנשים עבור ביצוע POC.
3. קוד python שנותן ניקוד לטיב התאמת נקודות המפתח על אדם בסרטון.

דיאגרמת הבלוקים של הפרויקט:



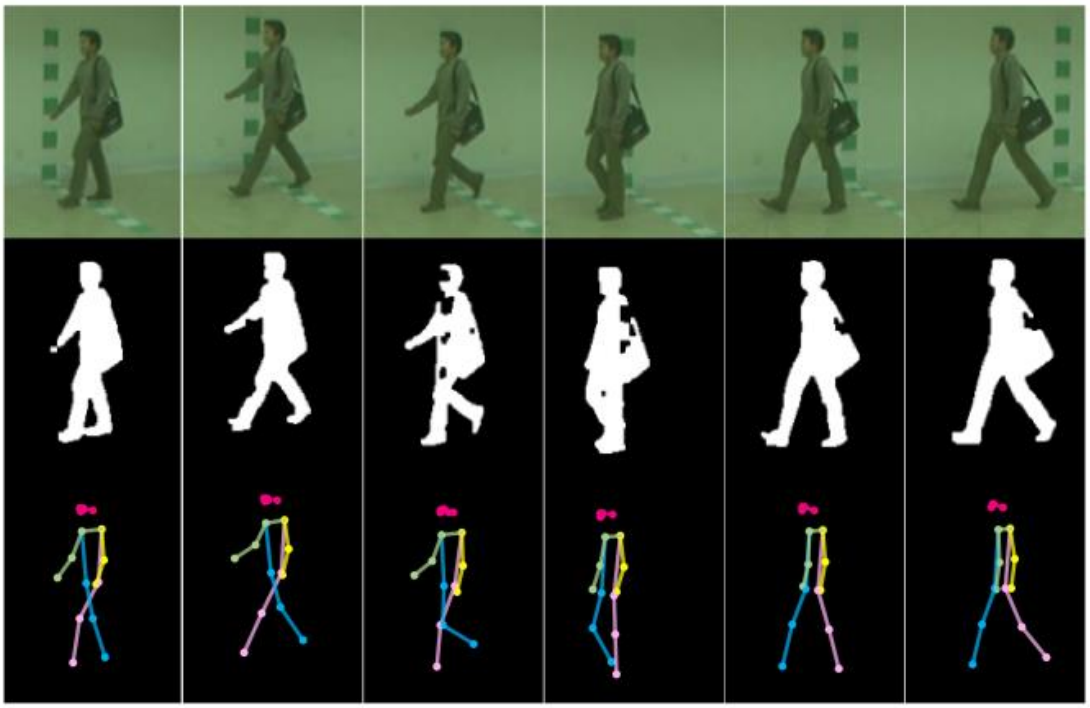
איור 1 - דיאגראמת בלוקים

1 הקדמה

קלסיפיקציה ע"פ gait recognition היא בעיה המקבלת תשומת לב גדולה בשנים האחרונות לאור מהפכת הלמידה העמוקה והראייה הממוחשבת. לבעיה זו יש גישות שונות, המתבססות על תיאור תנועה של אדם באופנים שונים ומגוונים, כגון:

צלליות- בדרך זו מתבצעת הפרדה בין גוף האדם בסרטון לרקע, מה שמביא לקבלת תמונה בינארית של האדם ביחס לרקע. חלוקת לחץ- בדרך זו מודדים את חלוקת הלחץ שמפעילות רגלי אדם על משטח באמצעות משטחי מדידה, מה שמאפשר זיהוי אנשים ע"פ דפוסי לחץ אלה.

קיימות עוד שיטות שונות. לכל שיטה יש את החסרונות שלה, לדוגמא שיטת הצלליות בעייתית מכיוון שקל לרמות אותה, מאחר ואדם הלובש לדוגמא מעיל גדול וכובע, יראה שונה משמעותית מאותו אדם בבגדים קצרים. לשיטת הזיהוי ע"פ דפוסי לחץ יש מגבלה משמעותית מאוד, מאחר והיא דורשת שהאנשים אותם נרצה לזהות ילכו על משטח ספציפי אותו המזהה יצטרך להציב, וגם יש חשיבות למשקל שאותו אדם סוחב וכו', לכן שיטה זו אינה מתאימה לדוגמא לצרכים בטחוניים.



איור 2 - טכניקות gait recognition שונות [4]

אנחנו בחרנו להתמקד בשיטה אחרת- pose estimation. שיטה זו מאפשרת ניתוח הליכה של אדם ע"פ נקודות על גוף האדם. בפרויקט שלנו בחרנו במודל ספציפי הנקרא MMPOSE, המוציא 17 keypoints על האדם ומציג אותו בצורת skeleton.

גישה זו מתמודדת בצורה טובה יותר עם מצבים בהם הלבוש של אדם משתנה, ומאפשרת תיאור תלת ממדי של אדם בהתבסס על תמונה דו מימדית, לכן יש לה יתרון יחסי על השיטות הנ"ל, אך יש לו גם חסרונות- כאשר כמות הפיקסלים שאדם תופס בתמונה מסוימת אינה מספיקה המודל לא מצליח לשערך את הנקודות. ישנה גם חשיבות לזוויות הצילום ולתאורה.

מטרות הפרויקט הן:

1. שיפור מודל ה- MMPOSE עבור מצבי צילום קשים- בפרט רזולוציה נמוכה, כאשר האדם בתמונה מאוד רחוק מהמצלמה/איכות צילום ירודה בה כמות הפיקסלים שיוצרים את תמונת האדם בפריים קטנה מדי להצלחת המודל המקורי.

   השיפור יתבצע באמצעות העברת הסרטון במודל object detection הנקרא Yolov7.

מודל זה, המבוסס על למידה עמוקה, מאפשר זיהוי אנשים בסרטון, גם במצבי קיצון קשים (בהן הMMPOSE לא עובד), ומאפשר חיתוך של כל אדם באמצעות bounding box מלבני מסביב לאדם.
ניקח כל מלבן מסביב לאותו אדם, ובאמצעות סופר רזולוציה (שיטה המאפשרת את הגדלת הרזולוציה של תמונה בהתבסס על למידת מכונה) נגדיל את מספר הפיקסלים המרכיבים את האדם, ורק אז נעביר ב-MMPOSE המקורי.

2. POC לקלסיפיקציה בין אנשים ע"פ 17 נקודות המפתח של ה-MMPOSE.
בדיקת ההיתכנות תתבצע על ידי רשת קלסיפיקציה מבוססת LSTM ו- Adam Optimizer, עם דאטה עבור סיווג בין שני אנשים.
3. מחקר על הצלחת המודל המקורי והמשופר בתנאים שונים- רזולוציה, זוויות צילום ביחס לקרקע, תאורה. לשם כך בנינו פונקציית score שלוקחת בחשבון אורכים וזוויות הגיוניים בגוף האדם, הורדת ניקוד על קפיצות מסוימות skeleton בין פריימים עוקבים בסרטון וכו'.

2 רקע תיאורטי

2.1 Gait recognition

Gait recognition [5], המוכר גם בשם human walking pattern recognition, הוא תחום ביומטרי המתמקד בזיהוי וניתוח של דפוסי הליכה אנושיים כדי לזהות ולהבדיל בין אנשים.

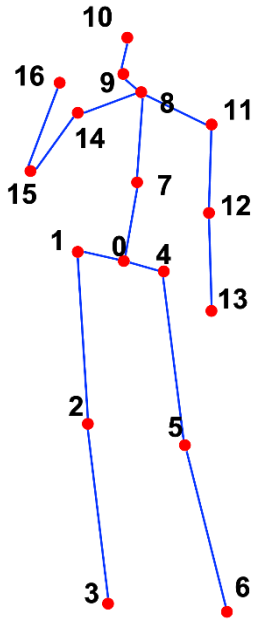
זיהוי אנשים לפי הליכה הוא תחום עתיק, שהיה קיים עוד בימי קדם ברומא וסין העתיקות- כשחיילים ושומרים הוכשרו לזהות את מפקדיהם וקצינים בכירים על ידי התבוננות בדפוסי ההליכה שלהם, מה שאפשר להם לזהות מתחזים או מרגלים.
כיום, התחום מפותח בהרבה, הוא התקדם להיות מדע כמותי, שמימושו הנפוץ ביותר הוא בתחום הראייה הממוחשבת. הופעתן של מערכות מבוססות וידאו (כמו מצלמות וידאו) בשנות ה-90 אפשרה איסוף של נתוני הליכה מקיפים יותר, מה שהוביל לפיתוח טכניקות ראייה ממוחשבת לזיהוי הליכה אוטומטי.

מספר שיטות הוצעו ופותחו לזיהוי הליכה, כל אחת משתמשת בטכניקות שונות לחילוץ וסיווג תכונות.

בין הגישות הרווחות כיום ניתן למנות :

- גישות מבוססות מודל- גישות אלו יוצרות סדרת פרמטרים סטטיים או דינמיים של הגוף באמצעות מעקב אחר רכיבי גוף כגון, רגליים, ידיים וראש (הערכת נקודות אלו נקראת pose estimation).
יש לגישה זו יתרון בכך שהוא מכיל יותר מידע על הצורה הפיזית של האדם, אך החיסרון הוא צורך ברזולוציה גבוהה מספיק בכדי לזהות במדויק את רכיבי הגוף.
- גישות נטולות מודל- גישות ללא מודלים מייצגות את תנועת ההליכה של האדם ההולך בהתבסס על צורות של צלליות או על התנועה המלאה של גופים אנושיים מבלי להתחשב במבנה הבסיסי שלהם. יש להם יתרון בעלויות חישוביות נמוכות מכיוון שהם לא צריכים לקבוע פרמטרים כדי לזהות את דפוסי התנועה המדויקים. החיסרון של גישות אלו הן חוסר הצלחה במצבים של שינוי פיזי של האדם המזוהה - תלבושות שונות, תספורת וכו'.
- קיימות גם גישות מבוססות היתוך (fusion), המשלבות בין שתי הגישות מעל.

בפרויקט שלנו אנחנו נשתמש בגישה מבוססת המודל **MMPOSE**, המעריך, דרך תמונות/סרטונים דו ממדיים, 17 נקודת המפתח במרחב התלת מימדי בנק' נבחרות על גוף האדם:



איור 3 - נקודות המפתח של מודל ה-MMPOSE

כאשר הנק' הן:

0-hip, 1-right hip, 2-right knee,

3-right foot, 4-left hip, 5- left knee,

6- left foot, 7- belly, 8- neck,

9- nose, 10- head, 11- left shoulder,

12- left elbow, 13- left hand, 14- right shoulder,

15- right elbow, 16- right hand.

2.2 רשתות נוירונים

רשתות ניורונים [6] פותחו על מנת לדמות דרך פעולה אשר דומה למערכת העצבים האנושית עבור משימות של למידת מכונה, על ידי כך שהיחידה החישובית במודל הלמידה מתנהג באופן דומה להתנהגות של הניורונים האנושיים - חיזוק/החלשה של קשרים בין הניורונים השונים.

בשנים האחרונות כוח החישוב של המחשבים גדל באופן משמעותי מאוד וזמינות מאגרי המידע השונים עלתה גם היא באופן משמעותי, כתוצאה מכך הביצועים של רשתות הניורונים בשנים האחרונות השיגו תוצאות טובות יותר בהשוואה להצלחת האדם בתחומים שונים כגון נהיגה אוטונומית, זיהוי תמונות ועוד.

ארכיטקטורה של רשתות נוירונים

כל שכבה של נוירונים מכילה את תאי הנוירונים אשר מחוברים בעזרת משקולות לנוירונים השונים משכבות אחרות.

מבנה זה של הרשת יוצר מסווג ליניארי:

$$(1) \quad f(x, W, b) = xW^T + b$$

כאשר

x – וקטור הקלט

W – מטריצת המשקולות

b – מקדם התאמה חופשי

f – וקטור התוצאה

כיוון שרוב בעיות הקלסיפיקציה אינן בעיות ליניאריות אנחנו מעבירים את הפלט של כל שכבה דרך פונקציית אקטיבציה אשר מכניסה חוסר ליניאריות לתוך דרך פתרון הבעיה. לדוגמא פונקציית האקטיבציה סיגמויד – σ :

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

חיבור של שכבות שונות כאשר המוצא של כל שכבה עובר דרך פונקציית אקטיבציה, יוצר את הארכיטקטורה השלמה של רשת הנוירונים.

RNN -Recurrent Neural Network

RNN [7] זוהי סוג של ארכיטקטורה אשר מיועדת עבור עיבוד מידע סדרתי.

בניגוד לרשתות נוירונים אחרות אשר מבצעות עיבוד על מידע מנקודות זמן שונה ללא תלות אחד בשני, *RNN* מסוגל למצוא תלות זמנית בהתבסס על המידע שהוכנס לרשת בעבר עבור מידע נוכחי שנכנס.

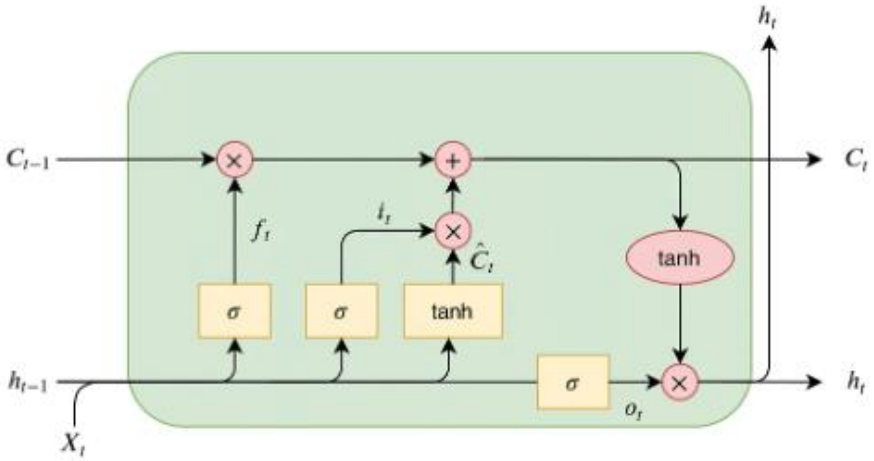
המאפיין העיקרי של סוג ארכיטקטורה זה הוא שיש חיבור בין שכבות קדמיות יותר אל שכבות מוקדמות יותר-

כלומר, מעין חיבור בקרה מעגלי אשר מאפשר לרשת למצוא קשרים ופיצ'רים עם תלות זמנית בין המידע שהוכנס ברגע זה ובין מידע מהעבר ובכך, להשפיע גם על המוצא עבור מידע עתידי שיכנס לתוך הרשת.

הבלוקים הבסיסיים שמהם מורכב ה*RNN* הם לרוב: *SRU (Simple Recurrent Unit)*

LSTM (long short term memory), או *GRU (Gated Recurrent Unit)*.

יחידות אלה מכילות פונקציות אקטיבציה פנימיות ומעצבות את מעבר המידע דרך רשת הנוירונים. בפרוייקט השתמשנו ביחידת ה-*LSTM*:



איור 4 - יחידת LSTM [8]

פונקציית *Loss*

פונקציית ה-*Loss* משמשת עבור מדידת הפער בין התווית האמיתית ובין התווית שרשת הנוירונים שיערה עבור המידע שהכנסנו אליה. המטרה של כל רשת נוירונים היא להקטין את ה-*Loss*, כלומר להגיע לתאימות אופטימלית בין הניבוי של הרשת למציאות. לרוב יתקבל קשר חזק בין הקטנת ה-*Loss* ושיפור הדיוק בזיהוי – זוהי מטרת הרשת.

BCEWITHLOGITSLOSS

זוהי פונקציית ה-*Loss* שהשתמשנו בה בפרויקט. פונקציית ה-*Loss* הזו היא שילוב בין שכבת אקטיבציה של סיגמויד ובין פונקציית ה-*Loss*: (*Binary Cross Entropy*) *BCELoss* [9]. שימוש בפונקציית *loss* זו נחשבת כמוצלחת במשימות של סיווג בינארי.

פונקציית ה-*Loss* מחושבת באופן הבא:

$$L = -w(y \cdot \log\sigma(x) + (1-y) \cdot \log(1-\sigma(x)))$$

כאשר:

x – וקטור הקלט

w – מטריצת המשקולות

y – וקטור התוצאה

L – *Loss* כולל

הערכים שפונקציה זו מחזירה קטנים יותר ככל שתוצאות ניבויי המודל קרובים יותר לאמת.

אימון ואופטימיזציה

כפי שציינו למעלה, רשתות ניורונים מכילות הרבה משקולות אשר מחברות בין חלקים שונים של הרשת וקובעות את הפלט שלה. המשקולות מתעדכנות במהלך תהליך האימון על מנת להקטין ככל הניתן את הערך שמתקבל מפונקציית ה-*Loss*.

תהליך זה נעשה על ידי שימוש בשיטת ה-*back-propagation* שבה הגרדיאנט של פונקציית ה-*Loss* מחושב לפי התלות בכל אחת מהמשקולות ברשת. חישוב זה נעשה בעזרת כלל השרשרת:

$$\frac{\partial g(f(x))}{\partial x} = \frac{\partial g(f(x))}{\partial f(x)} \cdot \frac{\partial f(x)}{\partial x}$$

לאחר מכאן, עבור עדכון אחרי i איטרציות נעדכן את המשקולות באופן הבא בעזרת הגרדיאנט:

$$w_{i+1} = w_i - \gamma_i \cdot \nabla L$$

כאשר:

w_{i+1} – המשקולות אחרי i איטרציות

γ_i – קצב הלימוד אחרי $i+1$ איטרציות

∇L – הגרדיאנט של ה$Loss$ לפי המשקולות

Adam Optimizer

Adam Optimizer [10] הוא מעין תוספת עבור הגרדיאנט הרגיל אשר הוכח להיות יעיל חישובית ומותאם לבעיות אשר מתמודדות עם מאגרי מידע גדולים.

בשיטה זו, לאחר כל איטרציה המשקולות של הרשת מחושבות באופן הבא:

$$w_{i+1} = w_i - \frac{\gamma_i}{\sqrt{A_i}} \cdot F_i$$

$$F_{i+1} = \beta_1 \cdot F_i + (1-\beta_1) \cdot \nabla L(w_i)$$

$$A_{i+1} = \beta_2 \cdot A_i + (1-\beta_2) \cdot \nabla L(w_i)^2$$

כאשר:

F_i – סכום משוקלל של הגרדיאנטים אחרי i איטרציות

A_i – סכום משוקלל של הגרדיאנטים בריבוע אחרי i איטרציות

Object detection

Object detection היא משימה בראיה ממוחשבת אשר שבה יש לזהות אובייקטים שונים במיקומים שונים בתמונה.
המשימה היא לא רק לסווג את האובייקט לפי קטגוריות שהוגדרו מראש אלא גם למצוא את המיקום שלהן בתוך הפריים של התמונה.

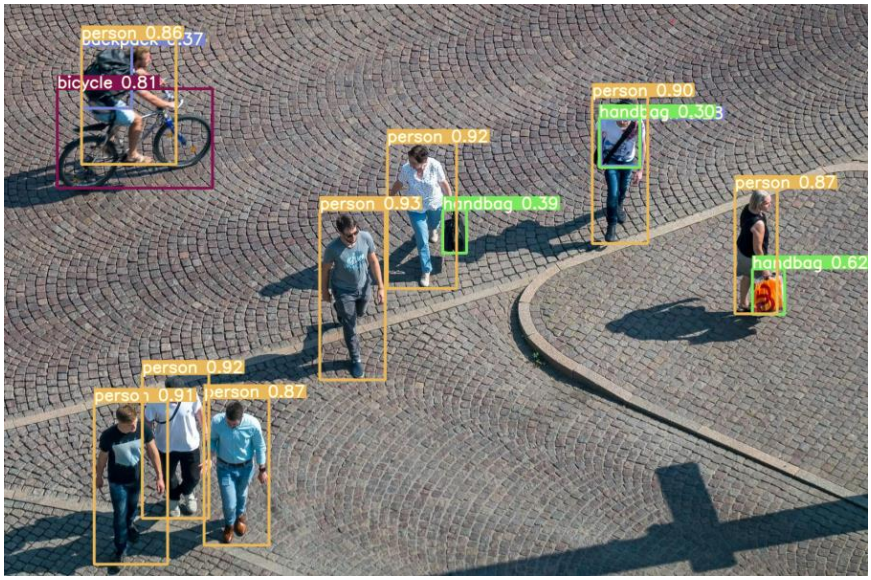
אלגוריתם ה-*object detection* לרוב נעשה בשני השלבים הבאים:

בשלב הראשון מזהים אזורים שונים בתמונה שיכולים להכיל אובייקטים אשר יעניין אותנו לזהות.

ישנן טכניקות שונות לביצוע מציאת האזורים כגון שימוש ברשתות קונבולוציה למציאת השטחים (*R-CNN*).

לאחר שהשטחים הרלוונטיים נמצאו, השלב הבא יהיה ביצוע סיווג של האובייקטים שנמצאים באותם אזורים,
ע"פ הקטגוריות השונות וימצא את המיקום המדויק יותר שלהם בתוך הפריים של התמונה. כיום שלב זה מבוצע בעיקר באמצעות אלגוריתמים שונים מבוססי למידת מכונה כאשר הדרכים הפופולאריות יותר משלבות גישות של למידה עמוקה.
לרוב עבור סיווג האובייקטים יעשה שימוש ברשתות קונבולוציה (*CNN*), ועבור מציאת המיקום המדויק יותר משתמשים בטכניקות רגרסיה אשר מייצרות *bounding box* סביב האובייקט.

הפלט של אלגוריתמי ה*Object detection* מכילים את התוויות של האובייקטים שזוהו בתמונה יחד עם הקואורדינטות של התחום שבו האלגוריתם זיהה את האובייקט. בעזרת המידע הזה על אותם אובייקטים שנמצאו אפשר לעשות עיבוד ייעודי למטרות שונות, כפי שאנחנו עושים בפרויקט זה עבור מציאת בני האדם בתמונה ושימוש במיקום שלהם בתמונה.



איור 5 - מוצא מודל ה- YOLOv7 [11]

סופר רזולוציה

סופר רזולוציה [12] זו טכניקה בעולמות הראיה הממוחשבת ועיבוד תמונות אשר משפרת את הרזולוציה של תמונות.
המטרה של טכניקה זו היא ליצור מתוך התמונה ברזולוציה הנמוכה תמונה ברזולוציה גבוהה אשר מכילה יותר פרטים ונראית חדה יותר.

הצורך בסופר רזולוציה עולה עקב סיבות שונות כגון צילום מלוויינים, שיפור איכות תמונות ישנות וכו'. טכניקה זו גם יכולה לעזור במצבים של צילום ממצלמות אבטחה כאשר נרצה לראות כמה שיותר מידע בתמונה על מנת להבין איזה אובייקטים שונים יש בתמונה, ובהתאמה לפרויקט הנ"ל כאשר רוצים למצוא באופן הטוב ביותר את בני האדם השונים בכל אחד מהפריימים על מנת לבצע על אותם בני אדם שזוהו עיבוד נוסף בעתיד.

אפשר לסווג את הטכניקות של הסופר רזולוציה לשתי טכניקות עיקריות:

הטכניקה הראשונה היא *Single Image Super Resolution (SISR)* - בטכניקה זו משפרים את הרזולוציה של תמונה בודדת בעלת רזולוציה נמוכה. בשיטה זו משתמשים בשיטות מבוססות למידת מכונה אשר מעריכות את התדרים הגבוהים שחסרים בתמונה שמעידים על שינויים תכופים יותר ופרטים מדויקים יותר בפריים, ומנסות להכניס את השינויים הללו על מנת להעלות את הרזולוציה של התמונה. טכניקות ה*SISR* הקלאסיות נעזרות בשיטות המבוססות אינטרפולציה.

הטכניקה השנייה היא *Multi Image Super Resolution (MISR)* - בטכניקה זו משפרים את הרזולוציה בעזרת כמה תמונות של אותו תרחיש המשמשות ליצירה של תמונה ברזולוציה גבוהה. בשיטה זו מצליבים מידע בין התמונות השונות על מנת לשערך את התדרים הגבוהים בתמונה שלא נמצאים בתמונה בעלת הרזולוציה הנמוכה.

לביצוע טכניקה זו משתמשים במקרים רבים ברשתות קונבולוציה על מנת ללמוד את דרכי המיפוי מרזולוציות נמוכות לרזולוציות גבוהות. מודלים אלו אומנו על מאגרי מידע גדולים אשר מכילים זוגות של תמונות ברזולוציה גבוהה וברזולוציות נמוכה על מנת שהרשת תוכל ללמוד איך לבצע את המעבר בין הרזולוציות.

Generative Adversarial Networks (GAN)

מודלים מבוססי *GAN* מכילים רשת אשר מבצעת גנרציה ורשת המבצעת אבחון. רשת הגנרציה לומדת לייצר תמונות ברזולוציות גבוהות ורשת האבחון תפקידה להבדיל בין התמונות ברזולוציה גבוהה שיוצרו על ידי רשת הגנרציה ותמונות ברזולוציה גבוהה אמיתיות. כאשר בשלב האימון של הרשת מכוונים לכך שרשת האבחון תבקר את רשת הגנרציה באופן כזה שהפרש בין מוצא רשת הגנרציה ובין התמונה האמיתית יהיה כמה שיותר קטן, כלומר מוודאת שהתמונה ברזולוציה הגבוהה אכן מציאותית דיו.

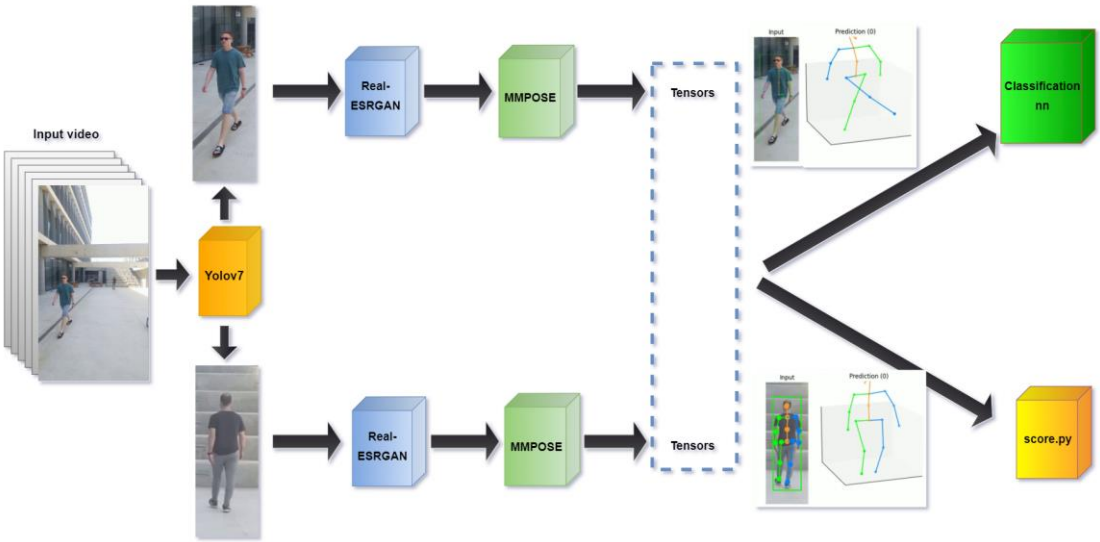
איור 6 - שיפור תמונה בעזרת SRGAN [13]

3 מימוש

מימוש הפרויקט מבוסס על 3 חלקים מרכזיים:

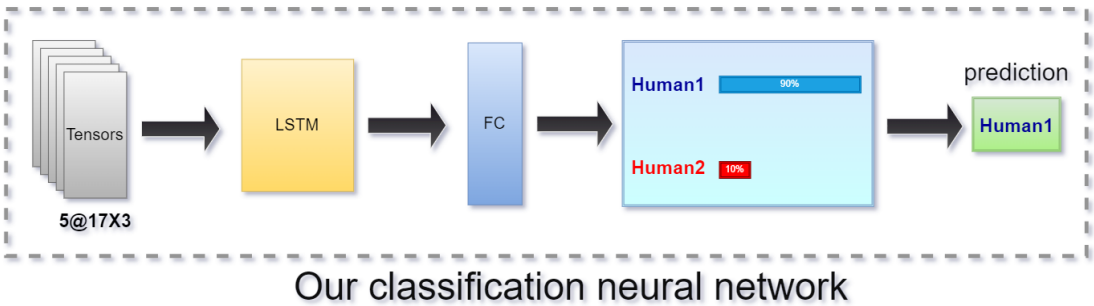
1. יחידה שתפקידה לעבד וידאו המכיל אנשים הולכים, ובסופה לקבל טנזורים המכילים את ה-keypoints שמתארים את גופו התלת ממדי של כל אדם בסרטון, בכל פריים.
   תפקיד יחידה זו היא לשפר את מודל ה-MMPOSE.
2. רשת קלסיפיקציה מבוססת למידה עמוקה, שתפקידה לקבל את הטנזורים ולסווג אותם לפי אנשים (הדאטה שלנו עבור האימון התבסס על שני אנשים שונים בשביל בדיקת ההיתכנות).
3. קוד בשפת python הנותן ניקוד להצלחת המודל במשימת מיקום הנקודות על גוף האדם המצולם. קוד זה מאפשר לנו לבצע מחקר על הצלחת המודל המקורי ושתי שיטות העיבוד שהצענו, בתנאים שונים (זוויות צילום שונות, תנאי תאורה וכו')

דיאגרמת הבלוקים של הפרויקט:



איור 7 - דיאגרמת בלוקים מפורטת של החלקים השונים בפרוייקט

כאשר רשת הקלסיפיקציה בנויה כך:



איור 8 - דיאגרמת בלוקים של רשת הניורונים

3.1 תיאור חומרה

המודל שלנו צריך לקבל סרטונים שצולמו מזוויות ספציפיות (שכן יש חשיבות להצלחת המודל כתלות בזווית, כפי שמופיע בהמשך בפרק התוצאות), והמימוש התוכנתי שלו מבצע עקיבה בהתבסס על כך שזווית צילום הסרטון סטטית. לכן, הסרטונים המוכנסים כקלט לסרטון דורשים שימוש בחצובה/כלי מייצב אחר שידאג למינימום תזוזות בזמן הצילום.

בנוסף, הקוד שלנו מניח שהמשתמש מחזיק בידו cuda-GPU, המשפר את זמן הריצה בכך שהוא מאפשר קוד הפועל בצורה מקבילית. אם למשתמש אין cuda, הוא יכול לשנות בצורה יחסית פשוטה את הקוד כך שיעבוד על ה-cpu, אך הוא מסתכן בזמן ריצה גבוה משמעותית.

3.2 תיאור תוכנה

הפרויקט שלנו נכתב ב- python 3.6.9 על שרת לינוקס. הוא מכיל את הבלוקים הקיימים המופיעים בתחילת הפרק, ומתבסס על מספר חבילות שהמרכזיות בהן הן numpy [14] ו-cv2 [15], המציעה פונקציות עיבוד תמונה ווידאו, ו-*PyTorch* [16] המאפשרת כלים לעבודה על ה-*GPU*, התורם מאוד לייעול סיבוכיות שלב של רשת הקלסיפיקציה.

עיבוד הנתונים נעשה ב-2 שיטות שונות, כאשר בשתי השיטות המודלים ששומשו זהים.

סדר הפעולות בקוד יתחיל בחלקים המשותפים לשתי השיטות:

1. קריאה לקוד main.py עם הארגומנטים: input video path, GPU_id.
2. main.py שולח את הסרטון ל-Yolov7 (לקוד detect.py) המחזיר את גבולות ה-bounding box (המלבן סביב כל אדם) עבור כל אדם ש-yolov7 זיהה. עבור מצב שבו יש יותר מאדם אחד אנחנו מבצעים עקיבה בסיסית סביב כל אחד מהאנשים, המתבסס על מיקום האדם בתמונה, שלא משתנה באופן משמעותי בין פריים לפריים. מטרת העקיבה לשייך בין אנשים בפריימים עוקבים ולאסוף בצורה יעילה וסקלבילית את המידע.
3. main.py מבצע, עבור כל אדם בסרטון חיתוך של תמונת הפריים לפי ה-bounding box שלו.
4. במידה והבאונדינג בוקס סביב האדם קטן יחסית, נרצה להגדיל את הרזולוציה שלו (כלומר את הפיקסלים המרכיבים את האדם). במצב כזה נשלח את תמונת חיתוך הפריים ל-Real ESRGAN (סופר רזולוציה), ונקבל תמונה גדולה יותר איתה ה-MMPOSE ידע להתמודד בצורה טובה יותר בהמשך.

כעת, מתבצע פיצול- השיטה הראשונה שלנו מבצעת את השלבים הבאים:

5. כל תמונה (שמתארת אדם ספציפי לאחר חיתוך ולאחר אולי SR) תומר לקובץ mp4 המכיל את התמונה כפריים אחד.
6. כל אחד מקבצי ה-mp4 ישלח ל-mmpose, שיבצע pose estimation ויחזיר קובץ טקסט המכיל את טנזורי ה-skeleton עבור כל אדם.

לעומת זאת, בשיטת העיבוד השנייה, מתבצעות פעולות מעט שונות לעיבוד התמונות של כל אדם:

5. ה-main ירפד באפסים את התמונות שמתארות כל אדם, לפי גודל התמונה המקסימלית המתארת את אותו אדם. התמונה תרופד בגבולות כך שהאדם נמצא במרכז התמונה.
6. כעת, כשכל התמונות של אותו אדם הן באותו גודל, ה-main יאחד את אותן תמונות לסרטון יחיד.
7. כל סרטון (שמתאר אדם ספציפי) ישלח ל-mmpose , שיבצע pose estimation ויחזיר קובץ טקסט המכיל את טנזורי ה-skeleton עבור כל אדם.

כלומר, ההבדל המרכזי הוא ששיטת העיבוד הראשונה שולחת את התמונות של כל אדם פריים-פריים למודול ה-*MMPOSE*, לעומת שיטת העיבוד השנייה שמבצעת פרוצדורות שמטרתן לשלוח ל-*MMPOSE* סרטון המכיל את הליכתו של כל אדם.

שיטת העיבוד השנייה נוצרה עקב תרחישים אליהם שמנו לב לאחר בחינת שיטת העיבוד הראשונה, שאמנם מטרתה לפתור את הבעיה של זיהוי אנשים ברזולוציה נמוכה, אך היא סובלת מקפיצות רבות, מאחר והיא מנסה לשערך נק' לכל אדם בכל פריים באופן בלתי תלוי. לעומת זאת הכנסה של סרטון שלם של ההליכה עוזר ל-*MMPOSE* לשערך *keypoints* בצורה חלקה יותר בין פריימים בשל התבססות על מיקומיהם בפריימים קודמים. בנוסף גם הריפוד באפסים ומירכוז תפקידו להקטין את הקפיצות בשל מיקום יחסית זהה של האדם בתמונה.

כעת, באופן נפרד מהקוד הנ"ל, קובץ הטקסט המכיל את הטנזורים ישלח לשני קודים נפרדים-
רשת הקלסיפיקציה בקוד nn.py **וקוד מתן הניקוד להצלחת ה-pose estimation על כל אדם** הנקרא score.py:

<u>רשת הקלסיפיקציה</u>

הרשת שבנינו מתבססת על ארכיטקטורה של *LSTM*. בעזרת סוג ארכיטקטורה זה נוכל לבצע ניתוח בין קלטים מנקודות זמן שונות ובכך למצוא פיצ'רים עמוקים יותר המעידים על תלות זמנית , אשר בפרויקט שלנו התלות הזמנית היא עבור מציאת תבנית הליכה ייחודית עבור אנשים שונים. ההבחנה בין תבניות ההליכה השונות נועדו על מנת לבצע את ההבדלה בין אנשים שונים על בסיס צורת ההליכה שלהם בלבד. וקטור הקלט של הרשת יהיה בגודל (5,17,3) ויוסבר בהמשך בחלק איסוף המידע ואימון הרשת מדוע זהו גודלו של הקלט.

ארגון והכנת הדאטא סט של הרשת

הדאטא סט צולם על ידינו עבור הבדיקות והניתוחים השונים שבוצעו בעזרת מצלמה של טלפון וחצובה בכדי לשמור על צילום סטטי.

הסרטונים שצולמו היו ברזולוציה של *3840x2160* וצולמו בקצב של כ-30 פריימים בשניה.

אותם סרטונים הוכנסו ל-*main.py* שהמיר את אותן תמונות אל טנזורי הנקודות שמודל ה-*pose-estimation* שהשתמשנו בו (*MMPOSE*) הוציא. נקודות אלה תיארו את ה-*skeleton* של כל אדם שהיה בתמונה. כלומר בעצם המרנו כל תמונה בת *3840x2160* פיקסלים ב-*RGB* לייצוג מרחבי (*x,y,z*) של 17 נקודות שונות בגוף עבור *N* אנשים בכל תמונה.

שייכנו כל אוסף נקודות *skeleton* לתווית האדם המתאימה לו, ועל בסיס אותן נקודות בלבד התבצע האימון והסיווג.

חלוקת הדאטא

חילקנו את הסרטונים השונים לשתי קבוצות שאינן חופפות באופן הבא:

Train – חלק זה קיבל 80% מהסרטונים. הסרטונים שהוקצו לחלק זה היו הסרטונים שהמודל התאמן עליהם, כלומר שעליהם הוא ביצע את האופטימיזציות הנדרשות עבור המשקולות של הנוירונים ברשת שלנו, שעל פיהן נקבעת תוצאת הסיווג. ווידאנו שבהגרלה של הסרטונים שהוקצו לחלק זה יהיה שיווין בין מספר הסרטונים של כל אחת מהמחלקות על מנת שלא יווצר מצב של הטיה ברשת למחלקה ספציפית.

Test – חלק זה קיבל 20% מהסרטונים. בחלק זה הניתוח של הרשת פעל ללא שינוי בערכי המשקולות השונים. זאת מכיוון שבחלק זה מטרתנו היא לבדוק את ההצלחה של האימון על גבי בסיס מידע שאינו תלוי בדאטא שעליו אומן המודל בעבר. על פי תוצאותיו של חלק זה יכלנו לאמוד את טיבן של תוצאות המודל ויכולות החיזוי שלו ובכך להעריך אם המודל הצליח להתמודד עם הבעיה באופן מספק.

אימון הרשת

על מנת לאמן את הרשת שלנו על הדאטא סט שלנו השתמשנו בספריית *PyTorch*. ספרייה זו הינה ספריית *open source* בתחום למידת מכונה בדגש על שימוש למטרות ראיה ממוחשבת. ספריה זו מספקת פיצ'ר של חישוב טנזורים מואץ בכרטיס הגרפי.

האימון התבצע על ידי בחירת השיטות ה-*hyperparameters* הבאים על מנת לאמן את המודל, כאשר הסברים על המשמעויות שלהם ניתן למצוא ברקע התיאורטי:

פונקציית ה-*Loss* שהשתמשנו בה היא *BCEWITHLOGITSLOSS*

ה-*optimizer* שהשתמשנו בו לאימון הרשת הוא *Adam optimizer* עם הפרמטרים: $\beta_1 = 0.9, \beta_2 = 0.999$

גודל ה-*batch* שהוכנס לרשת בכל פעימה הוא 5 פריימים – כלומר על מנת להצליח להבדיל בין האנשים השונים, הרשת חיפשה קשרים בין פעימות של חמישה פריימים עוקבים – 5 מטריצות בגודל *(17,3)* שהופכות לכניסה אחת בגודל של (5,17,3).

מספר ה-*epochs* שבוצעו היה 250. הדפסנו את התוצאות של כל אחד מה-*epochs* ושמרנו את המשקולות שלהם על מנת לבחור באיטרציה שהניבה את המשקולות של התוצאה האידיאלית.

Hidden size – 128. *hyperparameter* זה מייצג את כמות הנוירונים אשר נמצאים בשכבה הנסתרת של הרשת. הגודל של פרמטר זה משפיע על רמת המורכבות של הפיצ'רים שהרשת יכולה ללמוד בזמן תהליך האימון.

מספר הלייבלים לסיווג – 2. הרשת אמורה לסווג בין שני אנשים שונים שעליהם היא מתאמנת.

קצב הלמידה שבחרנו עבור הרשת אשר הניב לנו את התוצאות הטובות ביותר היה 0.001 .

<u>פונקציית מתן הניקוד</u>

פונקצייה זו נותנת ציון להצלחת המודל על כל אדם, באמצעות ממוצע משוקלל לפי סדרה של מעל ל-50 מבחנים הקשורים ליחסים בין אורכים בגוף האדם, זוויות בין מפרקים, קשרים בין יחסי אורך בין פריימים, קפיצות של זוויות הגוף בין פריימים וכו'.

כל מבחן ייתן למודל ציון בין 1-4. ציון זה נקבע ע"פ צפייה ב-GUI של פלט skeleton על מספר סרטונים נבחרים, והסקה מכך על טיב ההצלחה של המודל בכל מבחן.

כל פלט של פונקציית ה-main.py הנשלח (באופן נפרד מפעולת ה-main) אל פונקציית הסקור, מבצע את המבחנים הללו ובסיומן מקבל ציון בין 0 ל-1.

המבחנים שמימשנו מחולקים ל-4 סוגים, להלן דוגמאות למבחנים שמימשנו מכל סוג:

- מבחני יחס אורכים: היחס בין אורך פלג גוף שמאלי לפלג גוף ימני, היחס בין רוחב הכתפיים לגובה האדם.
- מבחני זוויות: הזווית בין הצוואר של אדם לקו המחבר בין שני כתפיו, הזווית בין הקו המחבר בין הצוואר של אדם למרכז המותניים שלו עם ציר ה-z.

- מבחני קפיצות אורכים: כל המבחנים היו יחס בין אורך מסוים בגוף בפריים הנוכחי ובפריים הקודם, חלקי היחס בין המרחק בין הצוואר לבטן בפריים הנוכחי ובעבר. חלוקה ביחס זה עזרה לנרמל את הטווח המתקבל במבחנים. האורכים אותם בדקנו הם קווי החיבור של skeleton המופיעים באיור 3 ברקע התאורטי, לדוגמא - אורך השוק, אורך יד תחתונה, יד עליונה, כתף וכו'.
- מבחני קפיצה בזווית- בדקנו את היחס בין זוויות מסוימות בין פריים נוכחי לפריים קודם, הזוויות שנבדקו היו אותן זוויות שערכן נבדק במבחני הזוויות הקודם.

בנוסף לממוצע המשוקלל, אפשרנו הורדה אגרסיבית יותר של ציון כאשר המודל נכשל במספר מבחנים גדול יותר.

הסיבה מאחורי הורדה זו היא כשנקודות שונות, בייחוד מצדדיו השונים של האדם, מתחילות לבצע קפיצות אשר באופן מובהק אינן נובעות מהתנועה הטבעיות של האדם (שינויים במהירויות לא אנושיות) אנחנו נרצה להעניש את הציון של תוצר זה בצורה משמעותית יותר.

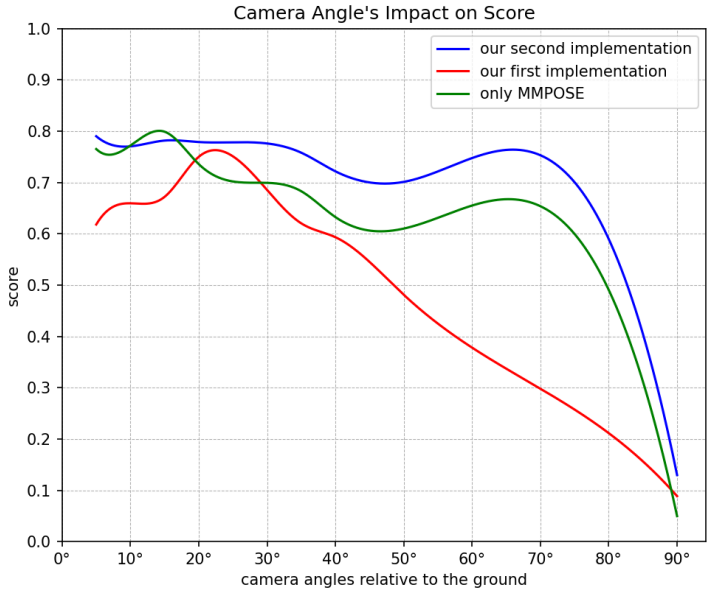
בחלק מזוויות הצילום קשה לראות ולנתח צד אחד של הגוף ומכאן יכולות להתעורר קפיצות באותו צד – תגובה זו של המודל נובעת מכך שאנו מאלצים אותו לנחש את מיקומי הנקודות על סמך מידע שלא עומד לרשותו.

לכן, אנחנו מחשיבים את חוסר הרציפות הזו כטעות פחות חמורה של המודל ויותר "טבעית", הנובעת מתוך מחסור באינפורמציה מהותית, זאת לעומת מצב בו שני צדדי skeleton מתנהגים בצורה לא הגיונית, כאשר אנחנו מצפים שלפחות צד אחד נראה בתמונה בצורה מספיק ברורה, כך שהמודל יוכל לספק בו שערוכי מיקום איכותיים. לכן, כאשר מצב זה לא מתקיים, נרצה להעניש בהתאם.

4 ניתוח תוצאות

השוואה בין ביצועי המודלים:

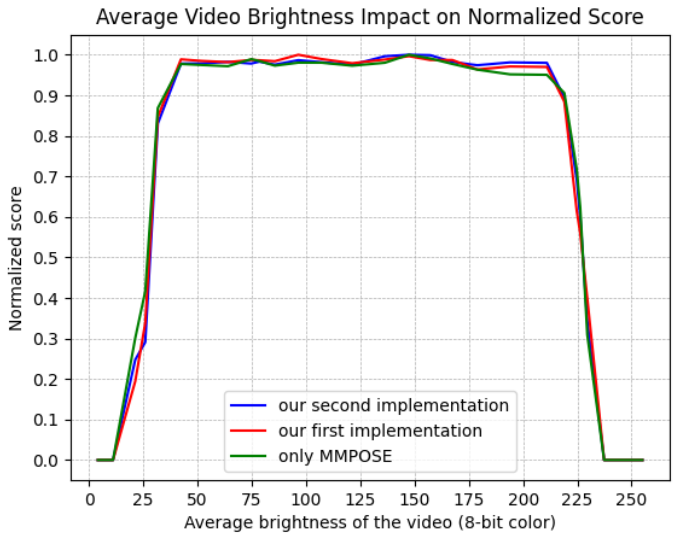
נרצה לנתח את הביצועים של שיטות העיבוד השונות. נתחיל תחילה מהצגת איכות המידע שמתקבל כאשר המידע מחולץ על ידי השיטות שונות. הרצנו את כל אחת מהשיטות על סרטונים אשר צולמו בזוויות שונות ביחס לאדמה ובדקנו בעזרת מודול ה-score.py את טיבן של התוצאות. קיבלנו את הגרף הבא:



איור 9 - ביצועי השיטות השונות עבור סרטונים שצולמו בזויות שונות

מהגרף נוכל לראות כי מלבד עדיפות קטנה של מודל ה-MMPOSE לבדו סביב ה-15 מעלות, השיטה השנייה שהשתמשנו בה הביאה תוצאות טובות יותר עבור שאר הזוויות שצילמנו. השיטה הראשונה שהשתמשנו בה הייתה עדיפה בכ-8.5% ב20 עד 30 מעלות, אך בזוויות גבוהות ראינו פערים משמעותיים בביצועים לעומת השיטה האלטרנטיבית וה-MMPOSE. מכאן, נוכל לומר כי אכן הצלחנו למצוא שיטה אשר שיפרה את טיב המידע שהצלחנו לדלות מתוך סרטון נתון.

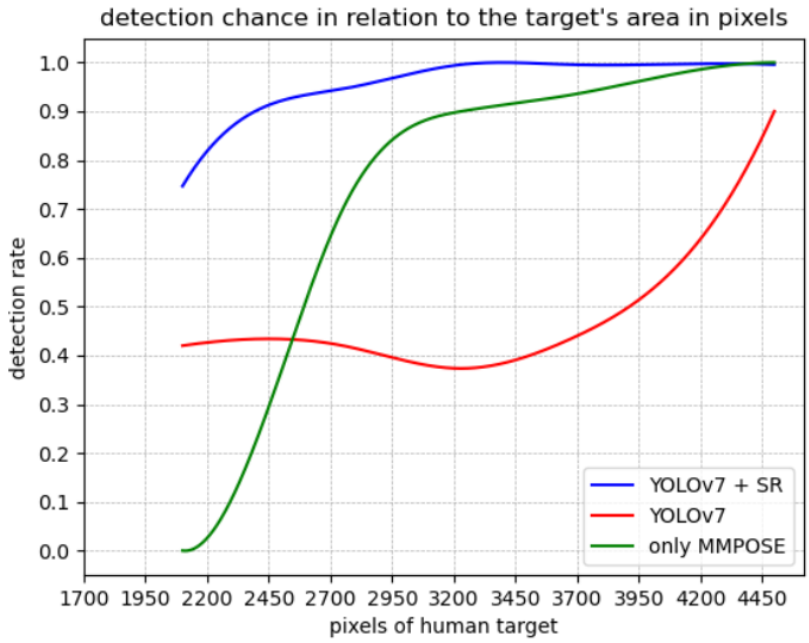
בדקנו את ההתנהגות של שלושת השיטות תחת תנאי הארה משתנים וקיבלנו את הגרף הבא:



איור 10 - ביצועי השיטות השונות עבור סרטונים שצולמו בתנאי תאורה שונים

מהגרף ניתן לראות כי עד כדי סטיות לא משמעותיות כל השיטות הציגו ביצועים דומים. ברוב הבהירויות המודל עבד בהצלחה ורק תחת תנאי תאורה קיצוניים ראינו השפעה ופגיעה ביכולת עיבוד הנתונים.

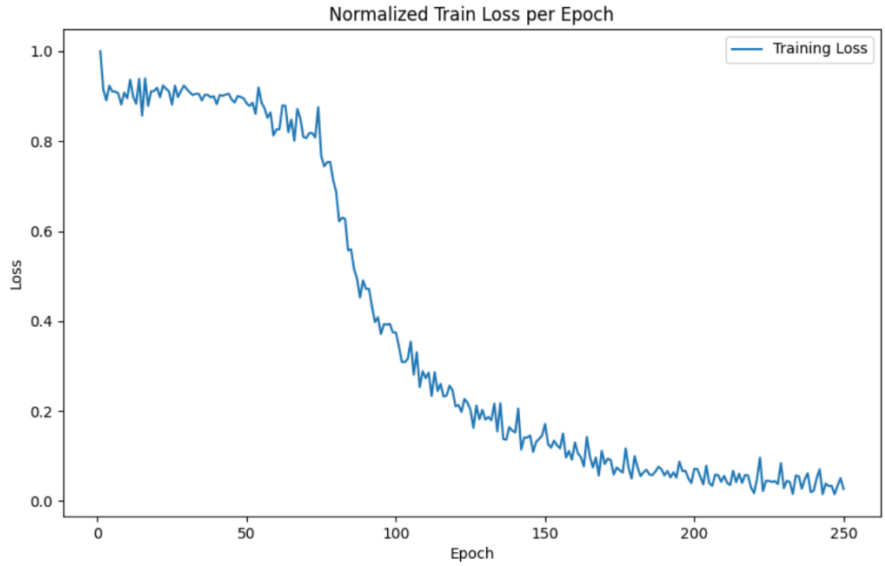
נרצה לבדוק את ההשפעה של שילוב מודלים שונים על הצלחת זיהוי אדם ושערוך ה-skeleton שלו. לשם כך בדקנו את אחוזי ההצלחה על סרטון שבו אדם הולך מרחוק ותופס כמות פיקסלים נמוכה. קיבלנו את התוצאות הבאות:



איור 11 - הצלחות הזיהוי עבור חיבורים של בלוקים שונים

<u>ביצועי רשת הניורונים:</u>

נבחן את תהליך האימון של הרשת:

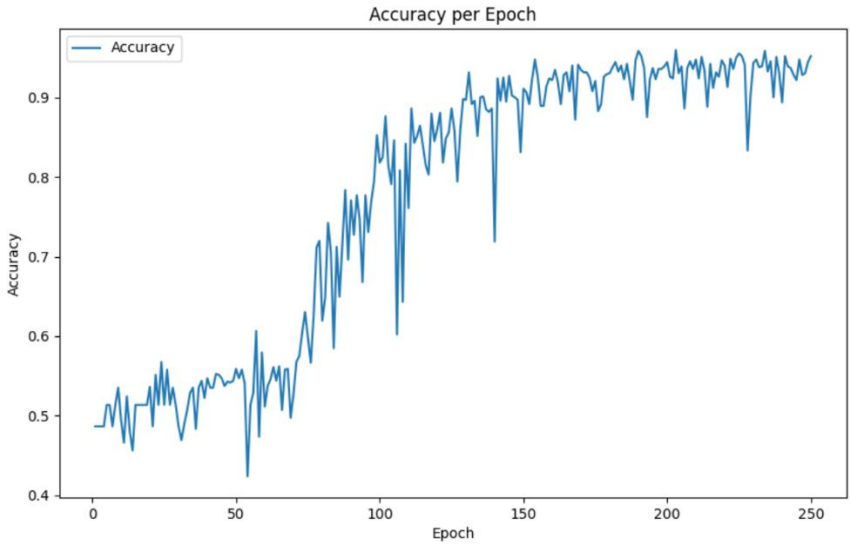


איור 12 - loss כתלות בepochs

אפשר לראות מהגרף שלאחר כ-75 epochs מתחילה להתרחש ירידה משמעותית יותר ב-loss וגודל זה יורד ככל שמספר ה-epochs גדל - מתאים להתנהגות שציפינו לקבל. לאחר 150 epochs אפשר לראות שה-loss

מתחיל להתכנס לערך נמוך יחסית שממנו הוא לא יורד באופן משמעותי, עד שעצרנו את האימון לאחר 250 epochs על מנת למנוע overfitting של המודל.

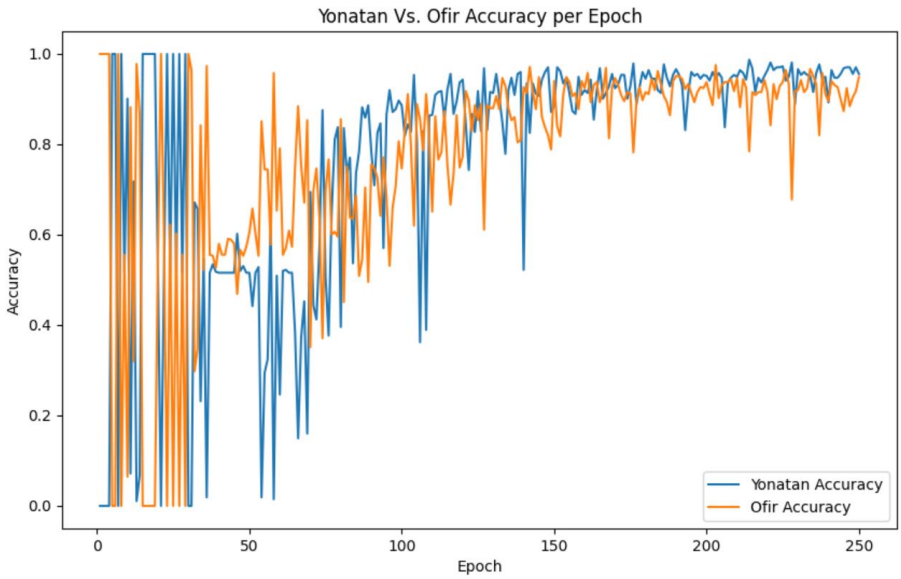
בהתאמה קיבלנו כי אחוז ההצלחה של הרשת על סט של ה-Test בכל אחד מהepochs היה:



איור 13 - אחוז הצלחת הרשת על ה-test set כתלות ב-epoch

גרף זה תואם למה שראינו מגרף ה-loss. סביב ה-epoch ה75 אנחנו מתחילים לראות עליה באחוזי ההצלחה של הרשת על הדאטא של ה-train set. האחוז הכי טוב שאנחנו מקבלים בתהליך זה הוא 95% אשר מעיד על הצלחה גבוה בהבדלה בין שני האנשים השונים.

בדקנו גם את ההצלחה בכל אחד מהמחלקות בנפרד על מנת לוודא שאין כאן הטיה לצידה של אחת מהמחלקות וקיבלנו את הגרף הבא:



איור 14 - אחוז הצלחת הרשת על ה-test set כתלות ב- epoch על כל אחד מהמחלקות

מתוך הגרף אכן אפשר לראות שאחוזי ההצלחה בסיווג לשני המחלקות קרובים זה לזה כפי שרצינו לראות על מנת לוודא שהרשת אומנה בהצלחה.

5 סיכום, מסקנות והצעות להמשך

המטרה של הפרויקט הייתה לקחת את מודל ה-*MMPOSE - pose estimation* ולשפר את יכולת רכישת ועיבוד המידע שלו על מנת להוציא מידע רב יותר ואיכותי יותר בהינתן תמונה או רצף תמונות. לאחר מכן הראינו היתכנות שבעזרת המידע שהתקבל מה*MMPOSE* אנחנו מצליחים לבנות מודל אשר יכול לסווג בין אנשים שונים על פי מאפיינים ביומטריים ודפוסים ביו-מכאניים זמניים של האנשים השונים.

בתהליך זה עבדנו עם שלושה מודלים קיימים מרכזיים – *Real-ESRGAN, MMPOSE* ו-*YOLOv7* כאשר כל אחד מהמודלים הביא לנו את השיפור הייחודי שלו בדרך לפתרון הבעיה. היה ניתן לראות במהלך העבודה על הפרויקט שיכולות זיהוי בני האדם של מודל ה-*YOLOv7* היו טובות בהרבה מהיכולות של *MMPOSE*, במיוחד כאשר היה הבדל משמעותי בין הגדלים של האנשים השונים בתמונה. לאחר שהמודל מוצא את האנשים אנחנו מבצעים חיתוך שלהם מתוך התמונה הכללית ומכאן מגיעה אחת התרומות המשמעותיות של השימוש במודל ה-*YOLOv7*. לאחר אותה פעולת חיתוך, אנחנו מקבלים כי ה-*MMPOSE* מזהה בצורה טובה יותר בחלק מהמקרים את האנשים וכתוצאה מכך אנחנו משיגים יותר מידע עבור אותם קלטים. עקב סיבות אלו הוחלט להשתמש במודל ה*YOLOv7* כיחידת עיבוד ראשונית עבור התמונה על מנת להצליח לתפוס כמות מידע רלוונטית גדולה יותר ובכך להניח את היסודות ליכולת של המערכת לזהות אנשים שונים ולאחר מכן לסווג ולהבדיל אותם אחד מהשני, כלומר להצליח לקשר בין אופיין הליכה מסוים לאדם ספציפי.

לאחר שראינו שה-*YOLOv7* מזהה אנשים ברמת סמך הגבוהה מרף מסוים, רצינו לשפר בחלק מהמקרים את איכות התמונה על מנת שה-*MMPOSE* יוכל להוציא ממנו את המידע הכי מדויק שהוא יכול, לכן ראינו כי השימוש במודל ה*Real-ESRGAN* אכן מצליח לענות לנו על הצורך ולהביא לנו יותר פרטים באופן כזה שמודל ה*MMPOSE* מזהה את הנקודות הרלוונטיות באדם שזוהה.

בעיתיות מסוימת שעלתה מהשימוש במודל ה*Real-ESRGAN* הייתה שמודלים מסוג זה בעצם מנסים לייצר מידע שלא קיים בתמונה וכתוצאה מכך עבור רזולוציות נמוכות מאוד קיבלנו לעיתים שחסרה גפה כלשהי באדם ועקב זאת התוצאות היו פחות מדויקות ביחס לגוף אנושי אמיתי. למרות חוסר הדיוק תוצאה זו הייתה מועילה לנו יותר מכיוון שללא תוספת זו מודל ה*MMPOSE* לא היה מצליח להשתמש בתמונה שהוא קיבל כקלט על מנת לבצע את העיבוד הרצוי עליה ולהוציא את מודל ה-*skeleton* עבור אותו אדם. אך במקרה הנ"ל, ה-*MMPOSE* זיהה את האדם אך נאלץ "להמציא" את מיקומי הנקודות החסרות.

נקודה נוספת לשיפור עקב השימוש במודל ה-*Real-ESRGAN* היא שמודל זה לא התאמן על תמונות של בני אדם באופן בלעדי. לכן, על מנת לשפר את הפעולה של המערכת ניתן להחליף את מודל זה במודל ייעודי אחר אשר מבצע סופר-רזולוציה אך אומן על תמונות של בני אדם. כתוצאה מכך נצפה שיתקבלו תוצאות חדות יותר ויציבות יותר של גבולות האדם בתמונה שיובילו לעיבוד ב*MMPOSE* שיניב לנו נקודות *skeleton* מדויקות יותר.

לאחר שאספנו את הדאטא של ה*skeleton* עבור מאגר סרטונים שצילמנו בו את עצמנו, רצינו לבדוק אם המידע שמערכת העיבוד שתכננו, אכן מניב תוצאות אשר בעזרתן נוכל להצליח להבדיל בין שני אנשים שונים על בסיס אופן ההליכה שלהם. רשת הקלסיפיקציה מבוססת *LSTM* שבנינו הראתה שבהינתן המידע (הן מבחינת איכות והן מבחינת כמות) שהבאנו לה ללמוד עליו הניבה תוצאות חיוביות ביותר. תוצאות אלו איששו באופן ברור את העובדה שבעזרת הדאטא שאספנו מהמערכת שלנו אפשר לבצע את ההבדלה שרצינו. כיוון שהראנו שבהינתן קבוצות של חמישה פריימים עוקבים ניתן למצוא קשרים ייחודיים עבור כל אדם על פי צורת ההליכה שלו אנחנו יכולים להניח שעבור מקבצי פריימים גדולים יותר נוכל לקבל קשרים ופיצ'רים ייחודיים יותר אשר יחזקו את יכולות החיזוי של המודל. יחד עם הגדלת מקבצי הפריימים שבניהם הרשת תמצא קשרים, נרצה להגדיל את כמות הנוירונים על מנת שנוכל באמת ללמד את הרשת למצוא כמות פיצ'רים גדולה יותר. לשם כך צריך, לאסוף כמות דאטא גדולה בהרבה ולעבד אותה על מנת לקבל את מיקומי ה*skeleton* של כל אדם בתמונה בשביל שנוכל להשלים את תהליך האימון של רשת הנוירונים כך שהיא תוכל למצוא קשרים עמוקים יותר ולהצליח לנבות בסיכויים גדולים יותר.

הקוד שייצרנו על מנת לנקד את התנהגות מוצא ה-skeleton אכן הביא לנו את תוצאות איכותיות על מנת לאמוד את טיב הפלט של הרשת במצבים השונים. לדעתנו השיפור של חלק זה יכול להיות ע"י שימוש באלגוריתמים של למידת מכונה ובכך

לנקד בצורה מדויקת יותר. ע"י כך נקבל את האפשרות לנתח ולנקד את תוצאת יחידת העיבוד שיצרנו בצורה טובה יותר ולקבל תוצאות מדויקות עוד יותר.

הצעות להמשך

מאחר והפרויקט עסק במשימה רחבה שמטרתה לזהות ולהבדיל בין אנשים שונים בתמונה, יש כיוונים רבים שבהם אפשר להמשיך ולפתח את הפרויקט:

שכלול רשת הקלסיפיקציה – ניתן לשנות את הרשת הנוכחית לרשת מורכבת יותר שבנויה על בסיס רעיון שהופיע במאמר – *FaceNet* [17] שבו, בניגוד לבעיה המוגדרת על כמות אנשים מוגבלת שרשת הקלסיפיקציה שלנו יודעת להתמודד איתה, בארכיטקטורה האלטרנטיבית יש את היכולת לחלק למספר קבוצות דינאמי שלא מחייב הגדרה ידנית. רשת זו מחפשת ולומדת פיצ'רים של תהליך ההליכה ובכך מזהה את ההבדלים והמרחק בין התוצאות הקודמות ובין האינפוט האחרון. כאשר תיווצר קבוצה חדשה בסבירות מספיק גבוהה עבור המודל (ניתן לבדיקה באמצעות קלאסטרינג קלאסי כדוגמת *k means*) נוכל לומר שזהו בן אדם חדש ולהגדיר אותו כקבוצה חדשה הניתנת לסיווג באיטרציות הקלט הבאות. בצורה זו נוכל לעבוד על כמות אנשים לא מוגדרת ולא נהיה תלויים בהגדרות שקדמו לאימון. לשם כך יהיה צורך לאסוף כמות מידע גדולה בהרבה על מנת שהמודל יוכל לעשות את אותן הבחנות אשר יספיק לנו להבדיל בין האנשים השונים.

עקיבה משופרת ואפשרות לצילום דינמי – במידה ונרצה לשפר את מגוון המצבים בסרטון שאיתו אנחנו יכולים להתמודד ולהמשיך לרכוש את המידע בו בצורה מאורגנת ויעילה על מנת לנתח בצורה איכותית את המידע בהמשך נוכל לשנות את שיטת העקיבה לשיטה שיודעת להתמודד טוב יותר עם תרחישים נוספים שהמערכת הנוכחית לא יכולה להתמודד איתם, כמו הסתרה זמנית של אדם על ידי אנשים או אובייקטים שונים. שינוי שיטת העקיבה יכול גם לאפשר לאלגוריתם לעבוד בתרחישים בהם המצלמה אינה סטטית ביחס לסביבה אלא לדוגמא מחוברת לרחפן שזז בכיוונים שונים על פי הצורך.

דרך העקיבה הראשונה שאפשר לפתח היא שיטה המבוססת על *optical flow*. הדרך השנייה שנציע למימוש העקיבה מתבססת על רשתות נוירונים. נוכל לייצר או לקחת מודל של רשת נוירונים אשר ידע לעקוב אחרי אובייקטים שונים בתמונה ובכך לשפר את כמות התרחישים שיכולים לקרות בסרטון ללא פגיעה בתהליך איסוף ועיבוד המידע שטמון בו.

שימוש במודלים אלטרנטיביים -לחקור ולבדוק על דרך אלטרנטיבית לייצור מידע שבעזרתה אנחנו ננתח את תבניות ההליכה של האנשים השונים. בפרויקט זה המחקר שלנו התבסס על מודל אשר מבצע *pose-estimation* בלבד – *MMPOSE*. ישנם עוד דרכים לקבל סוגי מידע שונים אשר יכולים לתרום לנו לתהליך העיבוד והסיווג בעתיד. שיטה נוספת שפועלת על מנת לפתור את הבעיה שפרויקט זה בא להתמודד איתה היא בעזרת טכניקת *fusion*. בעזרת שימוש בטכניקה זו אנחנו משלבים מידע שמתקבל דרך *pose-estimation* יחד עם מידע שהתקבל מ-*silhouette* של האדם בתמונה. בדרך זו אנחנו נשאף ללמוד ולאפיין פיצ'רים ייחודיים אשר יתקבלו דרך עיבוד הצלליות בשביל לקבל ניתוח מעמיק ומדויק יותר המתבסס על כמות מידע גדולה יותר אשר חולצה מתוך מאותו סרטון שצולם. לדעתנו שכלול השיטה להפקת המידע מהסרטון יהיה יעיל כאשר נרצה לבצע בעתיד את השדרוג לרשת הקלסיפיקציה שציינו קודם. הסיבה לכך היא שבגלל שנוכל לגזור מידע מגוון יותר מאותו קלט של המערכת, נוכל להצליח לסווג ולהבדיל בין התנהגויות שונות באופן איכותי יותר. יכולת זאת תעזור משמעותית לתפקוד הרשת במציאת הקשר בין אופי ההליכה של אדם נתון, ובין אופי ההליכה של האנשים השונים שעובדו בעבר על ידי המערכת, ובכך להחליט אם הוא ישות חדשה שצריך לקטלג אותה בתור קבוצה נוספת או שהוא שייך לאחת הישויות הקיימות.

המרת פונקציית הניקוד לרשת רגרסיה- עבור רשת מתן הניקוד, יש אפשרות לשפר את תפקודה באמצעות מימוש המבוסס על רשת עבור בעיות רגרסיה. רשת זו מאפשרת חיזוי ערכים מספריים רציפים, ע"פ מיפוי תכונות מהקלט. כך ניתן לתת ניקוד בין 0 ל-1 עבור קלטי טנזורים המכילים את נק' המפתח שהתקבלו ממוצא בלוק עיבוד הוידאו.

מאחר ובפרויקט שלנו מתן הניקוד התבסס על מחקר של ערכי תוצאות מבחנים ספציפיים שבחרנו, ומהם הושפע הניקוד מממוצע משוקלל, כלומר הנחנו שלכל מבחן יש חשיבות אחרת- יש צורך בהבנה מעמיקה יותר של איזה תכונות *skeleton* יש לקחת בחשבון וכמה משקל יש לתת לתכונה זו.

לכן - רשת רגרסיה מבוססת למידת מכונה יכולה להיות פתרון טוב מאוד לכך- ניתן לאמן עם כמות דאטה גבוהה יחסית המדורגת לפי ציון שהמשתמש ייתן, והרשת תלמד את הפרמטרים הרלוונטיים להצלחת וכשלון המודל, ובעצם תיצור "מבחנים" בעצמה, ותבין איזה פרמטרים בעלי חשיבות רבה יותר.

שיפור יכולת ניחוש מיקומי נקודות מפתח נסתרות במודל ה-*MMPOSE*- במצבי צילום מסוימים, בהן זווית המצלמה לא מאפשרת צפייה בחלק מהגוף (לדוגמא כאשר מצלמים אדם מצידו הימני, גפיו השמאליות נסתרות למצלמה), מודל ה-*MMPOSE* מתקשה לשערך את מיקום אותם מפרקים ובפועל מקבלים נק' במיקומים גרועים המובילים לזוויות ואורכי עצמות לא הגיוניים, מה שמעיד על כשלון יחסי של המודל.

נרצה לפתור את בעיית "ניחוש הנקודות" הלקויה בדומה לדרך הפעולה של מודלי שפה המתאמנים על השלמת משפטים [18].

במודלי שפה אלו, מתקבל קלט של תחילת משפט, והמודל צריך לנחש את המשכו, בהתבסס על מילים קודמות שנכתבו. על כל ניחוש כזה המודל מקבל ציון שממנו הוא לומד האם השלמת המשפט הייתה מוצלחת יותר או פחות.

כך- במצבים בהם לא ניתן לראות את כל 17 נקודות המפתח, מודל ה-*MMPOSE* מנחש את אותן נקודות. אנחנו נרצה "לתקן" את אותו ניחוש באמצעות מתן ציון לשערוך זה, מה שיוביל את ה-*MMPOSE* להעריך את הנק' בהמשך בהתבסס על אותו ציון של מיקומי הנקודות בפריימים קודמים, אורכי עצמות ידועים ועוד. כך, לאחר מספיק חזרות על רצף פעולה זה, צפוי שהמודל ידע לשפר את אופן החיזוי ולהעריך את מיקום נקודות המפתח בצורה טובה שתניב שגיאה קטנה יותר.

6 תיעוד הפרויקט

https://github.com/YonaRos/GaitRecognition

7 ביבליוגרפיה

[1] MMPOSE
https://mmpose.readthedocs.io/en/v0.29.0/demo.html#id10

[2] YOLOv7
https://github.com/WongKinYiu/yolov7

[3] Real-ESRGAN
https://github.com/xinntao/Real-ESRGAN

[4] Teepe, T., Gilg, J., Herzog, F., Hormann, S., & Rigoll, G. (2022). Towards a deeper understanding of skeleton-based gait recognition. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. https://doi.org/10.1109/cvprw56347.2022.00163

[5] Chuanfu Shen, Shiqi Yu, Jilong Wang, George Q Huang, and Liang Wang. A comprehensive survey on deep gait recognition: Algorithms, datasets and challenges. arXiv preprint arXiv:2206.13732, 2022.
https://arxiv.org/abs/2206.13732

[6] Aggarwal, C. C. (2018). *Neural Networks and Deep Learning*. Cham: Springer. ISBN: 978-3-319-94462-3

[7] Sherstinsky, A. (2020). Fundamentals of Recurrent Neural Network (RNN) and long short-term memory (LSTM) network. *Physica D: Nonlinear Phenomena*, *404*, 132306. https://doi.org/10.1016/j.physd.2019.132306

[8] LSTM unit image

https://www.projectpro.io/article/lstm-model/832

[9] Godoy, D. (2022, July 10). *Understanding binary cross-entropy / log loss: A visual explanation*. Medium.

https://towardsdatascience.com/understanding-binary-cross-entropy-log-loss-a-visual-explanation-a3ac6025181a

[10] Kingma, D. P. & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*
https://arxiv.org/abs/1412.6980

[11] Boesch, G. (2023, June 14). *Yolov7: The most powerful object detection algorithm (2023 guide)*. viso.ai. https://viso.ai/deep-learning/yolov7-guide

[12] Wang, X., Xie, L., Dong, C., & Shan, Y. (2021). Real-ESRGAN: Training real-world blind super-resolution with pure synthetic data. *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*.
https://doi.org/10.1109/iccvw54120.2021.00217

[13] Sciforce. (2022, January 12). *What's next for Gans: Latest techniques and applications*. Medium.
https://medium.com/sciforce/whats-next-for-gans-latest-techniques-and-applications-3be06a7e5ab9

[14]NumPy.

https://numpy.org/

[15] *Opencv-python*. PyPI.

https://pypi.org/project/opencv-python/

[16] PyTorch - About

https://web.archive.org/web/20180615190804/https://pytorch.org/about/

[17] Schroff, F., Kalenichenko, D. & Philbin, J. (2015). FaceNet: A unified embedding for face recognition and clustering.. *CVPR* (p./pp. 815-823), : IEEE Computer Society. ISBN: 978-1-4673-6964-0

https://arxiv.org/abs/1503.03832

[18] Mikolov, T., Chen, K., Corrado, G. & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *CoRR*, abs/1301.3781.

https://arxiv.org/abs/1301.3781