פרויקט סיום בקורס "בית הספר הישראלי לביומטריה השלישי"

מגישה שיר מאי רפפורט 318172467

בפרויקט זה אני אשווה, אעבוד ואבדוק את התכונה הביומטרית: פנים ולייתר דיוק- זיהוי פנים.

הערות חשובות:

* מצורף גם קובץ וורד של מסמך זה הואיל ושם ניתן לראות את האנימציה של חלק ב', **אנימציה של תהליך ה $Morphing$**
(אשר נמצאת בצד ימין של התמונה עם תשעת תמונות הפנים, **זו לא תמונה** גדולה של הפריים הממוסגר בריבוע לבן)
* הסבר על קבצי הקוד (של חלק א' +ב') נמצאים בסוף מסמך זה .
* יש לחפש בקבצי הקוד : $CHANGE\ HERE$ ולשנות את הנתיבים היחסיים לנתיבים התואמים לשלכם. הם יראו כך, למשל :

```
############# CHANGE HERE: update the paths for thresholds_path and scores_path #############
```

חלק א'

בחלק זה של הפרויקט בחרתי בשלושת המודלים/כלים שונים לזיהוי פנים הבאים והם : $Facenet,\ VGG-Face$ ו- $OpenCV$.

השתמשתי במאגר התמונות $CASIA-FaceV5$ אשר מכיל קבצים מסוג BMP של 500 נבדקים שונים. עבור כל נבדק קיימים חמישה קבצים אשר בהם הנבדק מופיע בכמה זוויות שונות ובתנאי תאורה שונים, כך שכל מודל/כלי נדרש להתמודד עם אתגרי וריאציה טבעיים.
נוכח התחשבות במשאבים של זיכרון, זמן ריצה ונפח חישוב- נבחרו מתוך המאגר כ-50 נבדקים ונוצרו 1,000 זוגות של תמונות הנבדקים, כאשר ישנם 500 זוגות של תמונות של אותו נבדק ($label=1$) ו- 500 זוגות אשר נוצרו באופן רנדומלי של נבדקים שונים ($label=0$).
על בסיס זוגות אלו הוערכו שלושת הכלים.

מעט על כל כלי/מודל: $Facenet$ ו- $VGG-Face$ הם מודלים מתקדמים מבוססי למידה עמוקה. שניהם מופעלים דרך ספריית $DeepFace$ ומחזירים מדד דמיון בין שני הפנים מבוסס $cosine\ distance$, כאשר כל תמונה מומרת לווקטור מספרי (embedding) שמייצג את תווי הפנים שלה בצורה מרחבית. לאחר מכן, עבור כל זוג תמונות, מחשבים את המרחק הקוסינוסי בין שני הווקטורים האלה (זה מדד של כמה שני הווקטורים מצביעים לאותו כיוון במרחב). ולכן *** ערך נמוך מעיד על דמיון גבוה וערך גבוה מעיד על דמיון נמוך (הערכים נעים בין 0 ל-2).
לעומתם, הכלי של $OpenCV$ מבוסס על אלגוריתם $LBPH$ שהיא גישה מסורתית, פשוטה וקלילה יותר, שאינה עושה שימוש בלמידה עמוקה אלא בחישוב טקסטורות מקומיות והמרתן לווקטור תכונות (כאשר $LBPH-Local\ Binary\ Patterns\ with\ Histograms$, כאשר אלגוריתם LBP ממפה כל פיקסל בתמונה לדפוס בינארי המבוסס על ערכי הבהירות של הפיקסלים השכנים שלו, לאחר מכן, האלגוריתם מחשב את ההיסטוגרמות של הדפוסים הבינאריים האלה, כלומר מונה את תדירות הופעתם בכל אזור של התמונה,
כך שההיסטוגרמות משמשות כייצוג של התכונות בפנים. כלומר, $LBPH$ היא טכניקה שמנתחת דפוסי תאורה מקומיים בפנים וממפה אותם להיסטוגרמות, מה שמאפשר זיהוי וסיווג יעיל של פנים במערכות ראיית מכונה.)

עבור כל מודל/כלי, נשמרו ארבעה קבצים תחת תיקיית $Results$:
$scores_X.csv$, $thresholds_X.csv$, $roc_X.png$, $optimal_threshol_X.csv$, $prediction_analysis_X.csv$
כאשר X מסמל את שם המודל/כלי.

בנוסף, נשמר קובץ בשם $optimal_thresholds.csv$, אשר מכיל את הספים האופטימליים עבור כל מודל. (מצורפים בהמשך הדו"ח)
המשמעות של סף אופטימלי היא נקודת איזון סבירה בין $FMR\ (False\ Match\ Rate)$ ל- $FNMR\ (False\ Non-Match\ Rate)$,
או במילים אחרות, ערך $EER\ (Equal\ Error\ Rate)$ שהוא הנקודה בה שיעור הדחייה השגויה שווה לשיעור האישור השגוי.
כלומר, הנקודה בה סכום הטעויות (FMR + FNMR) הוא המינימלי מבין כל הסכומים שבקובץ $thresholds_X.csv$.

ולכן, עבור כל כלי זיהוי, בחרתי 2 דוגמאות עבור תיוג נכון ו- 2 דוגמאות עבור תיוג לא נכון. ההרצות שבוצעו, התבצעו באותם תנאים עבור שלושת הכלים, על אותם זוגות של תמונות. עבור כל אחד מהם חושבו ערכי FMR ו- $FNMR$, במגוון ספים,
על בסיס נבנתה עקומת ה- ROC (אשר עליה יפורט בהמשך).

להלן ארבעת הדוגמאות של תוצאות ההרצות של כל מודל + הניקוד של כל זוג, אשר מופיע בקובץ $scores_X.csv$ הרלוונטי.
שתי הדוגמאות הראשונות הן של זוגות עם תחזיות נכונות, כלומר עם $label=1$, כלומר זוגות של תמונות השייכות לאותו אדם בפועל + המודל גם הצליח בזיהוי שלו($True\ Positive$) , שכן ערך ה $score$ שלהן נמוך מהסף האופטימלי שנבחר לאותו מודל.
לעומת זאת, שתי הדוגמאות האחרונות הן של זוגות עם תחזיות שגויות, כלומר עם $label=0$, כלומר מדובר בתמונות של שני אנשים שונים , אך המודל טעה וסיווג אותם כאותו אדם ($False\ Positive$) , שכן ערך ה־$score$ שלהן היה נמוך מהסף האופטימלי וגרם למודל לחשוב בטעות שמדובר באותו אדם.

דוגמאות אלו נבחרו במטרה להמחיש את אופי הביצועים של המודל ברמת הדימוי
*(בשילוב של הידיעה מי הוא הסף האופטימלי + הידע מ- *** נבחרו ה $score$-ים הרלוונטים עבור כל דוגמא)

$FaceNet$

ערך הסף האופטימלי - 0.412

$Correct$

להלן דוגמא מספר 1 והניקוד שלה - 0.000633166112525618

$112_2.png$

$112_3.png$

להלן דוגמא מספר 2 וניקוד שלה - 0.128744194684216

$359_2.png$

$359_3.png$

כפי שניתן לראות, המודל צדק וזיהה נכון בשתי הפעמים שמדובר באותו אדם, כלומר: התווית $label = 1$ + הניקוד, כלומר ערכי ה $score$ -ים, היו נמוכים מערך הסף האופטימלי 0.412 ולכן הזיהוי סווג נכון כ- $\boldsymbol{True\ Positive}$.

InCorrect

להלן דוגמא מספר 1 והניקוד שלה - 0.145590158395357

$390_3.png$

$047_2.png$

להלן דוגמא מספר 2 והניקוד שלה - 0.132197059797819

$379_1.png$

$057_2.png$

כפי שניתן לראות, המודל טעה בזיהוי בשתי הפעמים משום שמדובר באנשים שונים, כלומר: התווית $label = 0$ + הניקוד, כלומר ערכי ה $score$ - ים, היו נמוכים מערך הסף האופטימלי 0.412 ולכן המודל סיווג בטעות את הזוג כמתאים וזו טעות מסוג - $\boldsymbol{False\ Positive}$.

לאחר הרצת הכלי $FaceNet$ על זוגות התמונות שנבחרו, התוצאות (אשר נשמרו בקובץ $prediction_analysis_Facenet.csv$) הן:

category	count	percent
True Positive (TP)	419/1000	41.9
False Positive (FP)	76/1000	7.6
True Negative (TN)	424/1000	42.4
False Negative (FN)	81/1000	8.1
Accuracy (%)	**84.3**	-
Error Rate (%)	**15.7**	-

כפי שניתן לראות, המודל הצליח לזהות בצורה נכונה 419 מתוך 500 זוגות תואמים $(label = 1)$, מה שמוביל לשיעור $True\ Positive\ (TP)$ של 41.9%. בנוסף, הוא סיווג נכון כלא תואמים 424 מתוך 500 זוגות לא תואמים $(label = 0)$, כלומר $True\ Negative\ (TN)$ בשיעור 42.4%.

לעומת זאת, נרשמו 76 מקרים שבהם המודל טעה וחשב בטעות ששני אנשים זהים כשלא היו $False\ Positive\ (FP)$ בשיעור של 7.6%. כמו כן, ב־81 מקרים המודל לא זיהה נכון זוגות תואמים $False\ Negative\ (FN)$ בשיעור של 8.1%.

<u>סה"כ:</u>

שיעור הדיוק של המודל עומד על $\boldsymbol{Accuracy} = (TP\ +\ TN)/\ total = (419\ +\ 424)\ /\ 1000\ =\ 0.843 = \mathbf{84.3\%}$

ושיעור הטעות עומד על $\boldsymbol{Error\ Rate} = (FP + FN)/total = (76\ +\ 81)\ /\ 1000\ =\ 0.157\ =\ \mathbf{15.7\%}$

כלומר, נראה כי מודל זה מציג תוצאות סבירות עם אחוז דיוק כולל של 84.3%, נתון שמצביע על ביצועים טובים בממוצע.
היחס בין טעות מסוג FP לטעות מסוג FN מאוזן יחסית, כך שאין נטייה ברורה של המודל להעדיף זהירות יתרה או הקלות בזיהוי.
המודל שומר על רמת ביצועים יציבה גם מול טשטושים, שינויי תנוחה ותאורה, כפי שנראה מדוגמאות שנבחרו.

לפי התמונות של הזוג הראשון וגם של הזוג השני, נראה כי מוצגים אותם אנשים בהטיות וזוויות מעט שונות, עם תנאי תאורה דומים, מיקום כללי של הראש והגוף שמור, רקע קבוע והבעת פנים ניטרלית. הניקוד של שני הזוגות היה נמוך מאוד: 0.0006 ו- 0.1287 , מערך הסף האופטימלי $\mathbf{0.412}$.
ערכי ניקוד אלו מעידים על דמיון כמעט מוחלט ובנוסף על כך שהזיהוי היה מוצלח והמודל עמיד בפני שינויים מתונים בזווית הראש.

מנגד, שני הזוגות השניים אשר סווגו בטעות כמתאימים, על אף שמדובר בשני אנשים שונים, קבלו ניקוד גם כן נמוך - 0.145 ו- 0.132, אך זאת קרה הואיל וביחס לניקוד של זוגות ה $imposter$ האחרים, ערכי ניקוד אלו (אשר שקולים למרחק) היו נמוכים יחסית ולכן המודל זיהה בטעות התאמה בין זוגות אלו. כפי שניתן לראות, אכן התנאים החזותיים דומים- רקע כהה, תאורה יחסית חלשה, הזוג הראשון מביט שמאלה, הזוג השני מביט קדימה בזווית דומה. בנוסף המבנה הכללי של זוגות אלו דומה ולכן עלול להטעות.

לסיכום- ניתן להתרשם שהשונות הביומטרית לא הייתה בולטת מספיק עבור המודל $FaceNet$, מה שהוביל לטעות בזיהוי. דוגמאות אלו ממחישות כיצד המודל נוטה לתפקד היטב כאשר יש התאמה מלאה או כמעט מלאה בין תמונות של אותו אדם, אך עלול לטעות כאשר קיימים קווי דמיון כלליים בין אנשים שונים, גם אם לא מדובר באותו אדם בפועל.

$VGG - Face$

הסף האופטימלי - 0.541

$Correct$

להלן דוגמא מספר 1 והניקוד שלה – 0.421540012

$327_0.png$

$327_4.png$

להלן דוגמא מספר 2 והניקוד שלה - 0.455769278

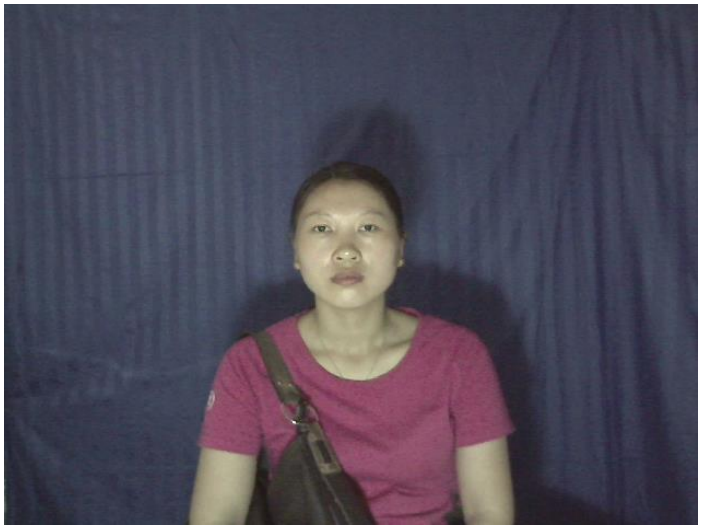
$140_1.png$

$140_4.png$

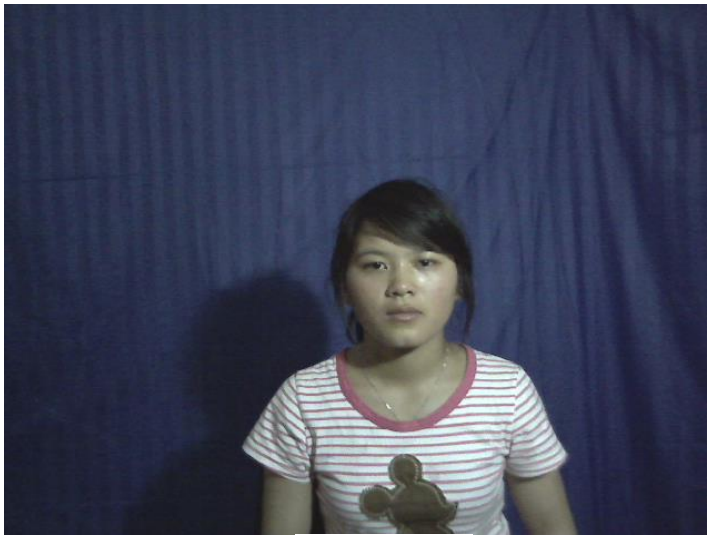
כפי שניתן לראות, המודל צדק וזיהה נכון בשתי הפעמים שמדובר באותו אדם, כלומר: התווית $label = 1$ + הניקוד, כלומר ערכי ה $score$ -ים, היו נמוכים מערך הסף האופטימלי 0.541 ולכן הזיהוי סווג נכון כ- $\boldsymbol{True\ Positive}$.

InCorrect

להלן דוגמא מספר 1 והניקוד שלה - 0.468751059468648



$346_3.png$



$456_0.png$

להלן דוגמא מספר 2 והניקוד שלה - 0.41709106130822

$301_3.png$

$111_2.png$

כפי שניתן לראות, המודל טעה בזיהוי בשתי הפעמים משום שמדובר באנשים שונים, כלומר: התווית $label = 0$ + הניקוד, כלומר ערכי ה $score$ - ים, היו נמוכים מערך הסף האופטימלי 0.541 ולכן המודל סיווג בטעות את הזוג כמתאים וזו טעות מסוג - $\boldsymbol{False\ Positive}$.

לאחר הרצת הכלי $VGG - Facve$ על זוגות התמונות שנבחרו , התוצאות (אשר נשמרו בקובץ $prediction_analysis_Facenet.csv$) הן:

category	count	percent
True Positive (TP)	410/1000	41
False Positive (FP)	21/1000	2.1
True Negative (TN)	479/1000	47.9
False Negative (FN)	90/1000	9
Accuracy (%)	**88.9**	-
Error Rate (%)	**11.1**	-

כפי שניתן לראות, המודל הצליח לזהות בצורה נכונה 410 מתוך 500 זוגות תואמים $(label = 1)$, מה שמוביל לשיעור $True\ Positive\ (TP)$ של 41%. בנוסף, הוא סיווג נכון כלא תואמים 479 מתוך 500 זוגות לא תואמים $(label = 0)$, כלומר $True\ Negative\ (TN)$ בשיעור 47.9%.

לעומת זאת, נרשמו 21 מקרים בלבד שבהם המודל טעה וחשב בטעות ששני אנשים זהים כשלא היו $False\ Positive\ (FP)$ בשיעור של 2.1% בלבד. כמו כן, ב־90 מקרים המודל לא זיהה נכון זוגות תואמים $False\ Negative\ (FN)$ בשיעור של 9%.

<u>סה"כ:</u>

שיעור הדיוק של המודל עומד על $\boldsymbol{Accuracy} = (TP + TN)/\ total = (410 + 479)\ /\ 1000 = 0.889 = \mathbf{88.9\%}$

ושיעור הטעות עומד על $\boldsymbol{Error\ Rate} = (FP + FN)/total = (21 + 90)\ /\ 1000 = 0.111 = \mathbf{11.1\%}$

כלומר, בהתבסס על נתונים אלו, נראה כי הביצועים של מודל זה מרשימים, עם שיעור דיוק גבוה במיוחד ו- $False\ Positive$ נמוך מאוד, נתון אשר מעיד על זהירות של המודל בזיהוי שגוי של אנשים שונים כאותו אדם. יחד עם זאת, ניתן לזהות נטייה מסוימת לפספס התאמות אמיתיות ($False\ Negative$ מעט גבוה יותר), כנראה במטרה להפחית את הסיכון לטעות בזיהוי של מתחזים. אפשר לומר שמדובר במודל שמרני, אשר מעדיף לא לזהות התאמה מאשר לטעות בזיהוי חיובי שגוי, תכונה שיכולה להיות חשובה במיוחד במערכות רגישות מבחינה אבטחתית. בהשוואה ל־$FaceNet$, מדובר בשיפור מובהק, בעיקר ביכולת של המודל להבחין בין אנשים שונים $(label = 0)$, מה שתורם ליציבות ואמינות גבוהה יותר בזיהוי.

בדומה לדוגמאות המודל הקודם- בתמונות של הזוג הראשון וגם של זוג השני מוצגים אותם אנשים בהטיות וזוויות מעט שונות, מיקום כללי של הראש והגוף שמור, רקע קבוע והבעת פנים ניטרלית, אך עם תנאי תאורה שונים. כמו שניתן לראות, הניקוד של שני הזוגות היה מעט נמוך: 0.4215 ו- 0.4557 מערך הסף האופטימלי- $\mathbf{0.541}$ והמודל צלח בזיהוי .נראה כי המודל הצליח להתמקד במאפיינים הבולטים ולהתעלם מהפרעות חזותיות סביבתיות.

מנגד, שני הזוגות השניים אשר סווגו בטעות כמתאימים, על אף שמדובר בשני אנשים שונים, קבלו ניקוד גם כן נמוך - 0.4687 ו- 0.4170, אך זאת קרה הואיל וביחס לניקוד של זוגות ה $imposter$ האחרים, ערכי ניקוד אלו (אשר שקולים למרחק) היו נמוכים יחסית ולכן המודל זיהה בטעות 'התאמה' בין זוגות אלו. כפי שניתן לראות, אכן התנאים החזותיים דומים- רקע בהה, תאורה יחסית חלשה, הזוג הראשון מביט שמאלה, הזוג השני מביט קדימה בזווית דומה. בנוסף, המבנה הכללי: דמיון בצבע העור, תנוחת הגוף, הבעת פנים דומה, או אולי מבנה עצמות לחיים, של זוגות אלו דומה ולכן עלול להטעות.

לסיכום, ניכר שהמודל $VGG - Face$ מבוסס על ייצוגים חזקים שמצליחים להתמודד עם שונות קלה בזוויות ובתאורה, אך במקרים מסוימים, תווי פנים כלליים או תנאי צילום דומים גורמים לו לבלבול בין זהויות שונות. הניתוח המייצג של ארבעת הזוגות מדגים ביצד המודל מתמודד עם משימות של התאמת פנים, ומספק מבט ממוקד על חוזקותיו וחולשותיו.

OpenCV

הסף האופטימלי - 12.653

Correct

להלן דוגמא מספר 1 והניקוד שלה – 5.69479720233948

$012_4.png$

$012_0.png$

להלן דוגמא מספר 2 והניקוד שלה - 5.55575720374049

$142_2.png$

$142_1.png$

כפי שניתן לראות, המודל צדק וזיהה נכון בשתי הפעמים שמדובר באותו אדם, כלומר: התווית $label = 1$ + הניקוד, כלומר ערכי ה $score$ -ים, היו נמוכים מערך הסף האופטימלי 12.653 ולכן הזיהוי סווג נכון כ- $\boldsymbol{True\ Positive}$.

InCorrect

להלן דוגמא מספר 1 והניקוד שלה - 11.7530262587008

$079_1.png$

$229_4.png$

להלן דוגמא מספר 2 והניקוד שלה - 11.61645491155

$214_2.png$

$174_1.png$

כפי שניתן לראות, המודל טעה בזיהוי בשתי הפעמים משום שמדובר באנשים שונים, כלומר: התווית $label = 0$ + הניקוד,
כלומר ערכי ה $score$ -ים, היו נמוכים מערך הסף האופטימלי 12.653 ולכן המודל סיווג בטעות את הזוג כמתאים וזו טעות מסוג -
$\boldsymbol{False\ Positive}$.

לאחר הרצת הכלי $OpenVC$ על זוגות התמונות שנבחרו , התוצאות (אשר נשמרו בקובץ $prediction_analysis_Facenet.csv$) הן:

category	count	percent
True Positive (TP)	430/1000	43
False Positive (FP)	64/1000	6.4
True Negative (TN)	436/1000	43.6
False Negative (FN)	70/1000	7
Accuracy (%)	**86.6**	-
Error Rate (%)	**13.4**	-

כפי שניתן לראות, המודל הצליח לזהות בצורה נכונה 430 מתוך 500 זוגות תואמים $(label = 1)$, מה שמוביל לשיעור $True\ Positive\ (TP)$ של 43%. בנוסף, הוא סיווג נכון כלא תואמים 436 מתוך 500 זוגות לא תואמים $(label = 0)$, כלומר $True\ Negative\ (TN)$ בשיעור 43.6%.

לעומת זאת, נרשמו 64 מקרים בלבד שבהם המודל טעה וחשב בטעות ששני אנשים זהים כשלא היו $False\ Positive\ (FP)$ בשיעור של 6.4% בלבד. כמו כן, ב־70 מקרים המודל לא זיהה נכון זוגות תואמים $False\ Negative\ (FN)$ בשיעור של 7%.

<u>סה"כ:</u>

שיעור הדיוק של המודל עומד על $\boldsymbol{Accuracy} = (TP + TN)/\ total = (430 + 436)\ /\ 1000 = 0.866 = \mathbf{86.6\%}$

ושיעור הטעות עומד על $\boldsymbol{Error\ Rate} = (FP + FN)/total = (64 + 70)\ /\ 1000 = 0.134 = \mathbf{13.4\%}$

כלומר, בהתבסס על נתונים אלו, ניתן להסיק כי מדובר במודל בעל ביצועים טובים למדי, במיוחד לאור העובדה שמדובר בכלי פשוט יחסית לשאר הכלים $FaceNet$ ו-$VGG - Face$. אחוזי הדיוק הכלליים גבוהים והאיזון בין FP ל־FN שמור. יחד עם זאת, ניכרת נטייה מסוימת של המודל לבצע יותר זיהויים שגויים של אנשים שונים כאילו היו אותו אדם (FP) , לעומת טעויות בזיהוי זוגות תואמים (FN) . כלומר, בניגוד ל- $VGG - Face$ למשל, מדובר במודל מעט פחות שמרני, שנוטה להעדיף זיהוי של התאמות גם במחיר של טעויות מסוג FP . באופן כללי, ניתן לומר כי זה הוא פתרון פשוט, יעיל וקל לשימוש, בעל יכולות טובות לזיהוי פנים, המתאים בעיקר למצבים שאינם קריטיים מבחינה אבטחתית.

בדומה לדוגמאות המודלים הקודמים- בתמונות של הזוג הראשון וגם של זוג השני מוצגים אותם אנשים בהטיות וזוויות מעט שונות, מיקום כללי של הראש והגוף שמור, רקע קבוע, תנאי תאורה שונים והבעת פנים ניטרלית. כמו שניתן לראות, הניקוד של שני הזוגות היה נמוך: 5.6947 ו- 5.5557 מערך הסף האופטימלי- $\mathbf{12.653}$, אך עדיין המודל צלח בזיהוי .נראה כי המודל הצליח להתמקד במאפיינים הבולטים ולהתעלם מהפרעות חזותיות סביבתיות.

מנגד, שני הזוגות השניים אשר סווגו בטעות כמתאימים, על אף שמדובר בשני אנשים שונים, קבלו ניקוד גם כן נמוך - 11.7530 ו- 11.6164, אך זאת קרה הואיל וביחס לניקוד של זוגות ה $imposter$ האחרים, ערכי ניקוד אלו (אשר שקולים למרחק) היו נמוכים יחסית ולכן המודל זיהה בטעות 'התאמה' בין זוגות אלו. כפי שניתן לראות, אכן התנאים החזותיים דומים- רקע בהה, תאורה יחסית חלשה, דמיון בצבע העור. ולכן עלולים להטעות את המודל.

לסיכום, המודל $OpenCV$ מדגים יכולת טובה יחסית בזיהוי התאמות, אך במקביל רגישות לדמיון חזותי כללי בתמונות $imposter$. ייתכן ששימוש באלגוריתמים מתקדמים יותר או שילוב של מאפיינים נוספים (מעבר למרחק פשוט) יוכל לשפר את הדיוק של מודל זה במצבים גבוליים. כמו כן, בהשוואה למודלים אחרים נראה כי הוא מצליח להבחין היטב בזוגות תואמים, אך עדיין מועד לטעויות כאשר מדובר בזוגות לא תואמים עם קווי דמיון כלליים.

גרף ה – ROC :

ראשית, אסביר את השתלשלות האירועים והצעדים אשר מובילים לבניית גרף זה:

1. יצירת זוגות:

- נבחרו **50 נבדקים אמיתיים מתוך 500 הקיימים במאגר $CASIA - FaceV5$.
- עבור כל נבדק, נוצרו 10 זוגות פנים מאותו אדם $(label = 1)$ + נוצרו 10 זוגות רנדומליים בין נבדק זה לבין אנשים אחרים $(label = 0)$.

כלומר, סה"כ נוצרו 500 זוגות תואמים + 500 זוגות לא תואמים = 1000 זוגות.

*אלו נשמרו בקובץ $pairs.csv$.

2. חישוב ציוני התאמה $(scores)$:

- לכל זוג תמונות הופעל הכלי $FaceNet, VGG - Face$ או $OpenCV$ וחושב מרחק קוסינוסי או $score$ אחר, תלוי בכלי:
  - ב־ $FaceNet/VGG - Face$ חושב מרחק קוסינוסי (קטן יותר = דמיון גבוה יותר).
  - ב־ $OpenCV\ (LBPH)$ חושב מרחק מבוסס־היסטוגרמות (קטן יותר = דמיון גבוה יותר).

*ניקודים אלו נשמרו בקובץ $scores_X.csv$ יחד עם ה $label$.

3. חישוב ספים ו - $FMR/FNMR$:

- נבדקו 50 ערכי $thresholds$ שונים, על פני טווח סביר של ערכי $score$ שנמצאו בפועל:
  - עבור הכלים $FaceNet/VGG - Face$ הספים היו מ־0.1 עד 1.0
  - עבור $OpenCV$ הספים היו מ-5 עד 30.
- עבור כל סף t נבדק: האם כל זוג נחשב כ־"אותו אדם" או "שני אנשים", לפי האם $score < t$ או $score \geq t$.
- בהתאם ל- $label$ + לתוצאת ההשוואה הוגדרו TP, FN, TN, FP באופן הבא:

TP (True Positive)	זיהוי נכון של אותו אדם → $score < t \ + \ label = 1$
FN (False Negative)	פספוס של אותו אדם → $score \geq t \ + \ label = 1$
TN (True Negative)	זיהוי נכון של מתחזים → $score \geq t \ + \ label = 0$
FP (False Positive)	טעות זיהוי של מתחזים כאותו אדם → $score < t \ + \ label = 0$

ומתוכם חושבו:

- $FMR = FP / (FP + TN)$: אחוז מההשוואות בין אנשים שונים שהמודל זיהה בטעות כאותו אדם, שגיאה מסוג $false\ accept$.
- $FNMR = FN / (FN + TP)$: אחוז מההשוואות בין תמונות של אותו אדם שהמודל זיהה בטעות כשונות, שגיאה מסוג $false\ reject$.

*התוצאות נשמרו בקובץ $thresholds_X.csv$.

4. בניית גרף ROC :

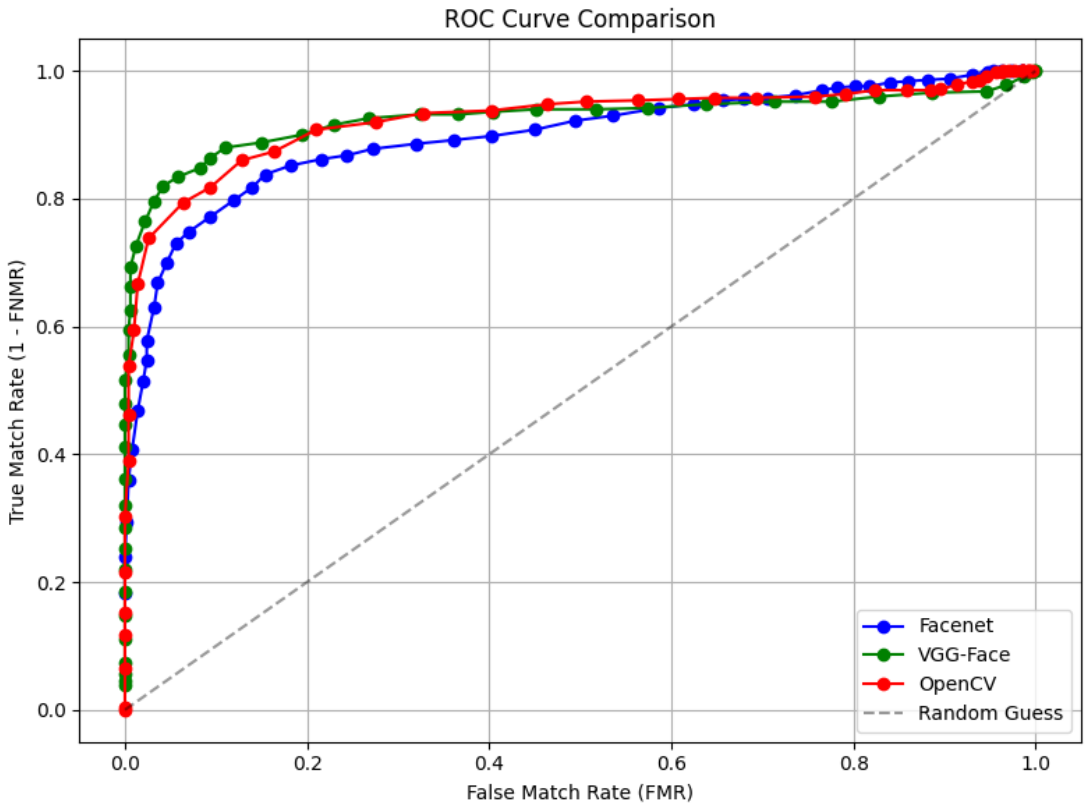
- לכל סף t נוצרה נקודה בגרף, כאשר הקורדינטות שלה הן : $x = FMR \ , \ y = 1 - FNMR$.
- התקבלו העקומות להשוואה בין הכלים.

**השאיפה הייתה לייצר קובץ מאוזן עבור ניתוח $FMR/FNMR$ ו- ROC , כדי לא להטות את המודל לטובת מחלקה מסוימת (כמו התחזויות לעומת התאמות אמיתיות). אם הייתי בוחרת בכל ה500 נבדקים, היו כ 10,000 זוגות אשר היו מובילים לסה"כ 30,000 בדיקות בתאמה וזה כמובן היה לוקח המון זמן ומשאבים.

בהתאם לצעדים אלו, כל נקודה בעקומת ה־ROC מייצגת סף אחר. כך ניתן לראות כיצד שינוי הסף משפיע על הביצועים: כלומר,
כאשר הסף נמוך מדי : מתקבל FMR גבוה מאוד(~ הרבה שגיאות מסוג $false\ accept$).
כאשר הסף גבוה מדי : מתקבל $FNMR$ גבוה (כלומר פספוסים של זיהויים אמיתיים).

עקומת ה־ROC מאפשרת להעריך בצורה חזותית את הרגישות והדיוק של כל כלי ביומטרי לאורך כל טווח הספים האפשריים, ולא רק בנקודה אחת. זה הוא כלי חיוני להשוואה בין אלגוריתמים בתחום הזיהוי.
ואת זאת ניתן להבין באמצעות הגרף באופן הבא:
<u>עקומת ROC אופטימלית היא זו שמתקרבת ככל האפשר לפינה השמאלית־עליונה של הגרף</u>, כלומר, משיגה שיעור זיהוי נכון גבוה $1 - FNMR$ גם כאשר שיעור ההתאמות השגויות FMR נמוך. ככל שהעקומה גבוהה יותר ונמצאת בצד שמאל של הגרף, כך ביצועי המודל נחשבים לטובים יותר. עקומה אשר נמצאת מעל עקומות אחרות מצביעה על כך שבאותם תנאי סף, המודל מצליח להבחין טוב יותר בין זהויות זהות לשונות.

ובניסוח יותר פרקטי:
FMR ≈ 0 : אין כמעט זיהויים שגויים של אנשים זרים
FNMR ≈ 0 : אין כמעט פספוסים של אנשים אמיתיים



ולכן, מן ההשוואה בין שלוש העקומות המוצגות בגרף עולה כי המודל $VGG - Face$ מפגין ביצועים עדיפים ביחס לאחרים, במיוחד באזורי סף רגישים שבהם שיעור ההתאמות השגויות נמוך(FMR בין 0 ל־0.2) .ניתן לראות כי העקומה הירוקה, המייצגת את $VGGF - Face$, נמצאת ברוב התחום מעל לעקומות של $FaceNet$ ו־ $OpenCV$, דבר המעיד על כך שביחס לכל סף שנבחן,
המודל מצליח לזהות נכון יותר את הזהויות האמיתיות תוך שמירה על רמת שגיאות התחזות נמוכה. ממצא זה משתלב עם ניתוח הדוגמאות הספציפיות שהוצגו קודם וממחיש את חוזקות המודל לא רק ברמת הזוגות הבודדים אלא גם בהתנהגות הגלובלית שלו.

בנוסף, ניתן להבחין בכך שהמודלים מגיבים באופן שונה כאשר נדרשת רמת דיוק גבוהה במיוחד. לדוגמה :
בציר האיקס בטווח שבין 0 ל־0.05 , כלומר כאשר שיעור הזיהויים השגויים FMR נמוך מאוד, ניתן לראות שהעקומה של $FaceNet$ נמצאת נמוך משמעותית ביחס לשתי העקומות האחרות, כלומר, במצב של סף מחמיר, המודל מתקשה בזיהוי נכון של זוגות אמיתיים, ומציג שיעור $FNMR$ גבוה. תופעה זו מעידה כי המודל "נשבר" מהר מבחינת רגישות כאשר מחמירים את הדרישה לספציפיות.
לעומתו $VGG - Face$ מצליח לשמור על רמת זיהוי נכונה גבוהה ($sensitivity$ גבוהה) גם באזורים אלו ומציג עקומה גבוהה ויציבה יותר, ואילו $OpenCV$ שומר על איזון הדרגתי בין שני המדדים.

התנהגויות אלו ממחישות כי לכל אחד מהמודלים יש אופי אחר מבחינת האיזון בין רגישות ($sensitivity$) לספציפיות ($specificity$).
העקומה של $FaceNet$ מדגישה נטייה לשמור על ספציפיות על חשבון רגישות, בעוד $VGG - Face$ מצליח לאזן היטב ביניהן, לפחות בטווחים נמוכים של FMR . ניתוח זה, חשוב לצורך בחירת המודל המתאים בהתאם לדרישות היישום – מערכת שבה דרושה סבילות נמוכה לטעויות זיהוי שגוי, תעדיף ככל הנראה מודל ששומר על רגישות גם בסף נמוך, בעוד שמערכת שמעדיפה להימנע מ- $False\ Matches$ עלולה לבחור בגישה מחמירה יותר גם במחיר של פספוס.

חישוב הספים האופטימליים:

לאחר חישוב ערכי ה־FMR וה- $FNMR$ לאורך טווח של ספים שונים עבור כל כלי, עבור כל כלי נבחר הסף אשר יוצר איזון מיטבי בין טעות מסוג אחד (FMR) לטעות מסוג שתיים $(FNMR)$. הטבלה שלהלן מציגה את הסף שנבחר עבור כל כלי זיהוי פנים, ואת ערכי השגיאה שהתקבלו עבור סף זה:

Tool	Optimal Threshold	FMR	FNMR
$FaceNet$	0.412	0.154	0.162
$VGG-Face$	0.541	0.042	0.18
$OpenCV$	12.653	0.128	0.14

לסיכום חלק זה, ניתן לראות כי שלושת הכלים שנבחנו הציגו תפקוד שונה בהתמודדות עם זיהוי פנים בתנאים מגוונים.
ניתוח עקומות ה־ROC בשילוב עם הדוגמאות הספציפיות שנבחרו והנתונים המספריים בטבלת הספים האופטימליים,
מדגישים את ההבדלים בין הכלים הן ברמת הדימוי הבודד והן בהתנהגות הכללית של כל מודל לאורך טווח רחב של ספים.
מהטבלה עולה כי המודל $VGG-Face$ הצליח להציג את שיעור ה-FMR הנמוך ביותר (0.042), גם אם במחיר של $FNMR$ מעט גבוה יותר (0.18), ואילו $FaceNet$ הציג איזון פחות מוצלח בין שני המדדים, עם ערכים גבוהים יחסית של FMR ו־$FNMR$ גם יחד.
לעומתם, $OpenCV$ אמנם פשוט יותר מבחינת מנגנון הפעולה, אך הצליח לשמור על איזון מספרי יחסית הדוק בין שני סוגי השגיאות $(FMR = 0.128, FNMR = 0.14)$ מה שמעיד על יציבות בביצועים בסף שנבחר.
תובנות אלו תורמות להבנת היתרונות והחסרונות היחסיים של כל מודל ומהוות בסיס חשוב לבחירה מושכלת של כלי או גישה להמשך, כפי שיבוא לידי ביטוי בחלק הבא של הפרויקט.

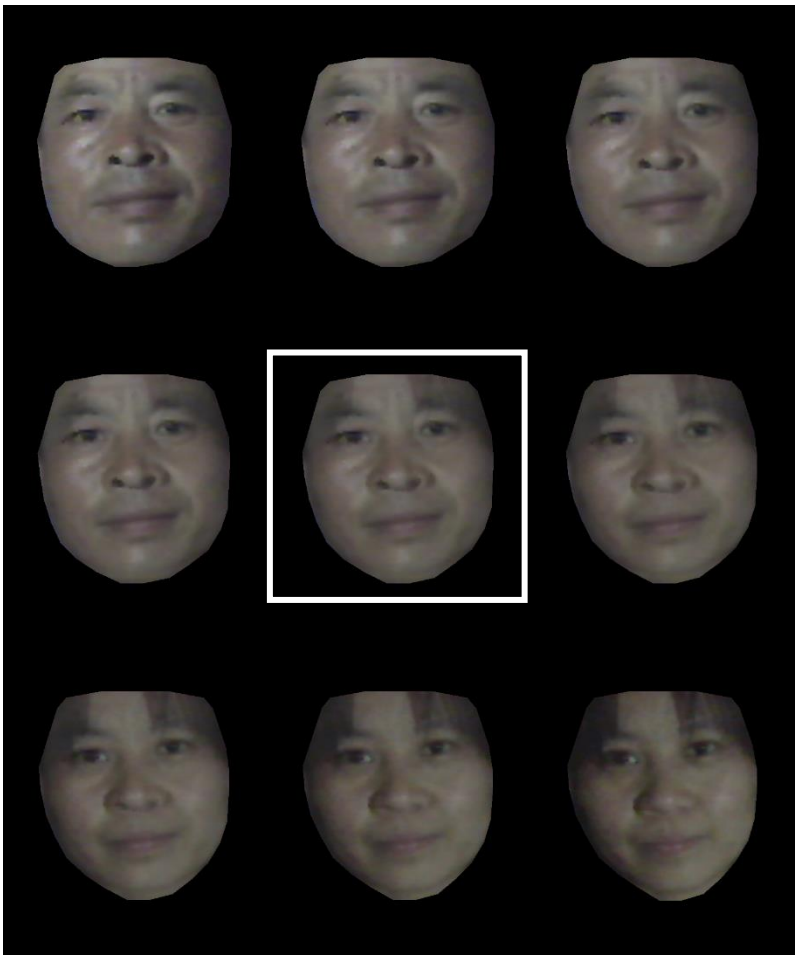
חלק ב'

בחלק זה של הפרויקט בחרתי באפשרות א- $\boldsymbol{Morphing\ Attack}$. בעקבות תוצאות חלק א', ביצעתי את ההתקפות על המודל שהפיק את התוצאות הטובות ביותר והוא $\boldsymbol{VGG-Face}$, כפי שניתן היה לראות מעקומת ה $\boldsymbol{ROC}$.
את האנשים בחרתי באופן רנדומלי ואת מספר הפריימים בחרתי כך שיהיו מספר אי זוגי של פריימים על מנת לבחור את הפריים האמצעי (אשר ממוסגר בריבוע לבן) לבדיקת ה $Morphing\ Attack$. להלן תוצאות הדוגמאות בהן בחרתי:
(מזכירה שוב, כי התמונה הימנית הגדולה, <u>היא לא תמונה</u>, **זו אנימציית $\boldsymbol{Morphing}$ שניתן לצפות בה בקובץ וורד של דו"ח זה)**

$\boldsymbol{M1 = Male\ To\ Woman}$

$207_0.png$

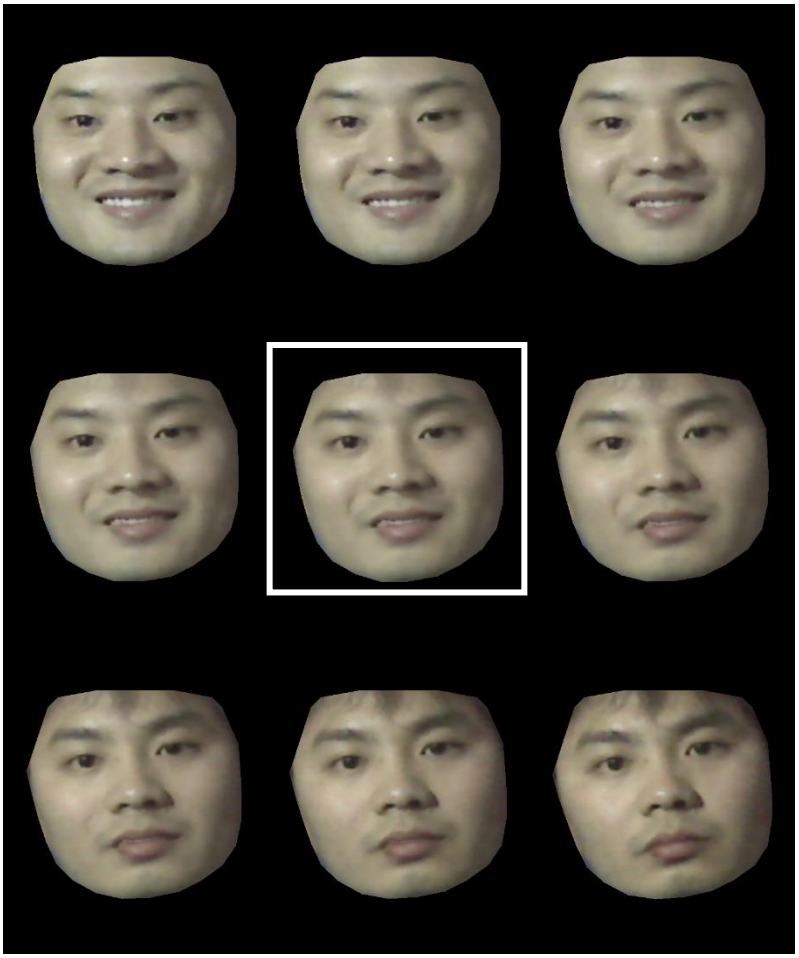
$079_0.png$



$\underline{\boldsymbol{M2 = Male\ To\ Male}}$

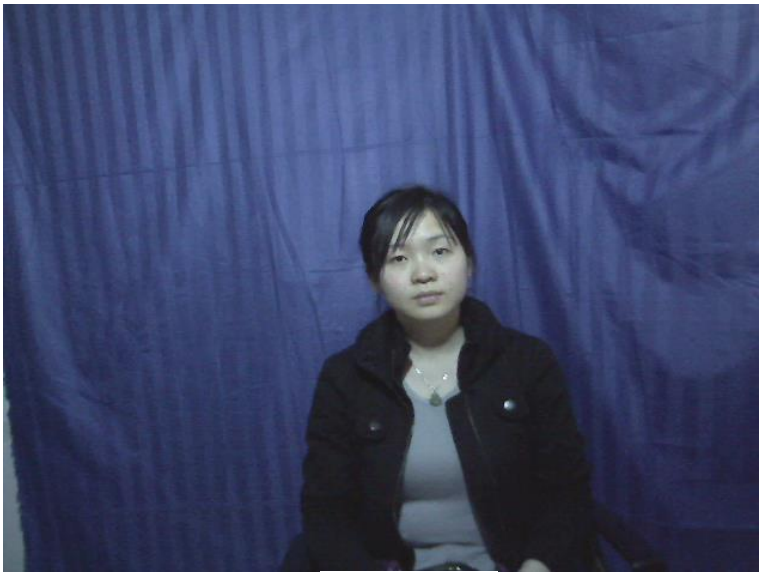
001_1.png

149_0.png

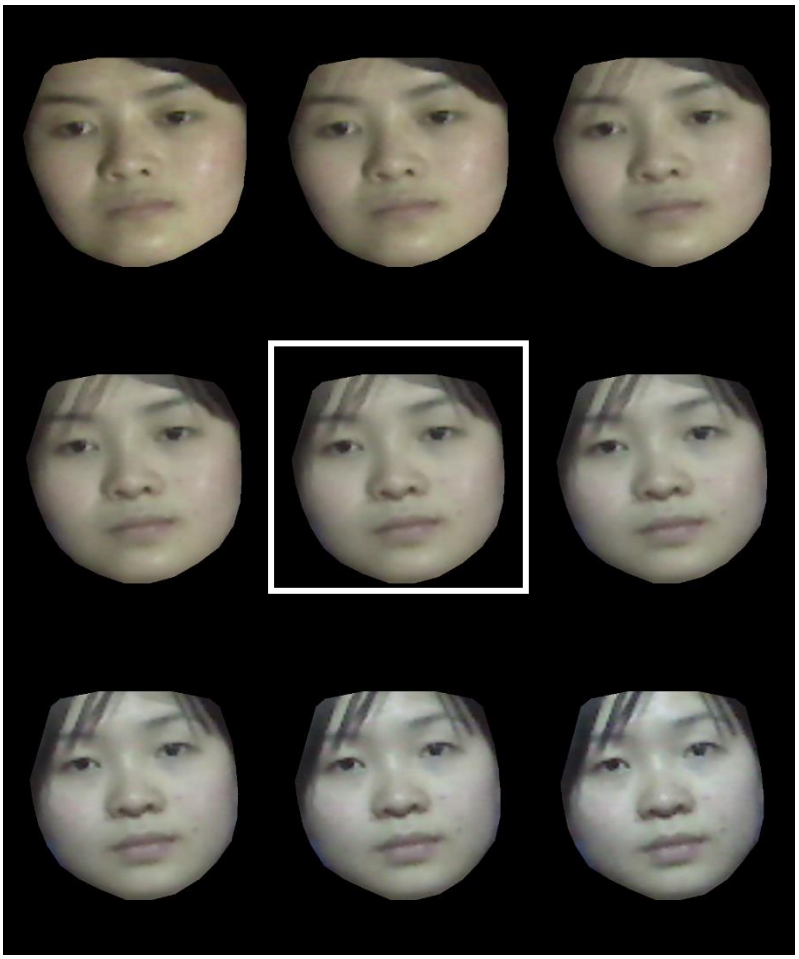


$\underline{M3 = Woman\ To\ Woman}$

136_1.png



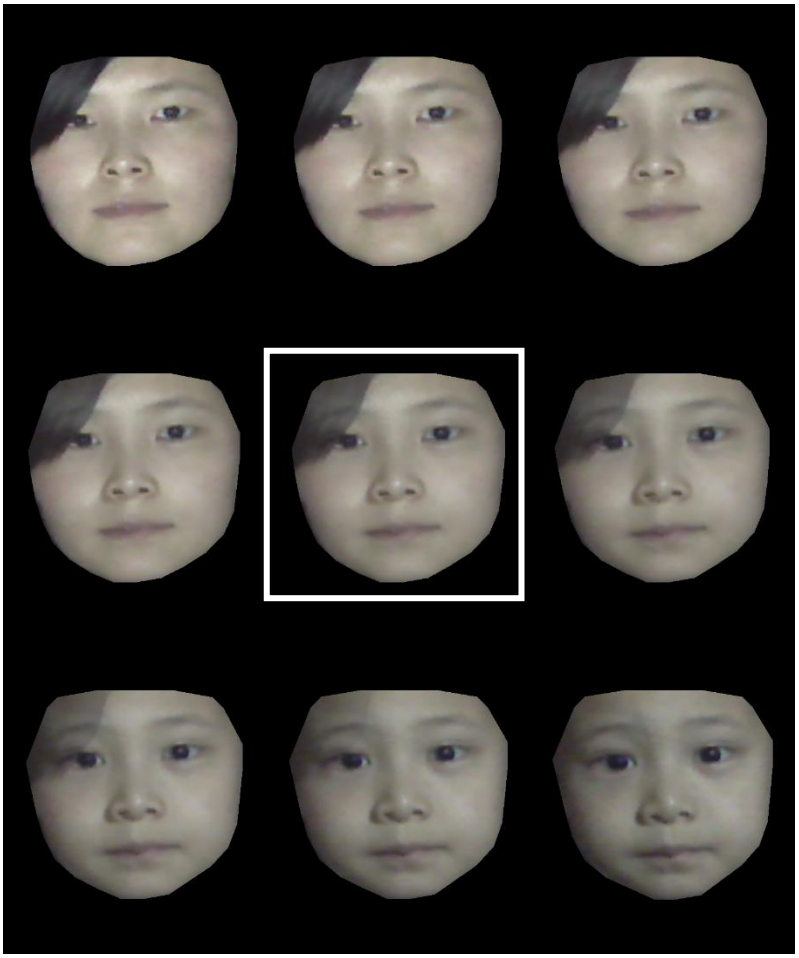
063_1.png



$\underline{\boldsymbol{M4 = Woman\ To\ Child}}$

318_1.png

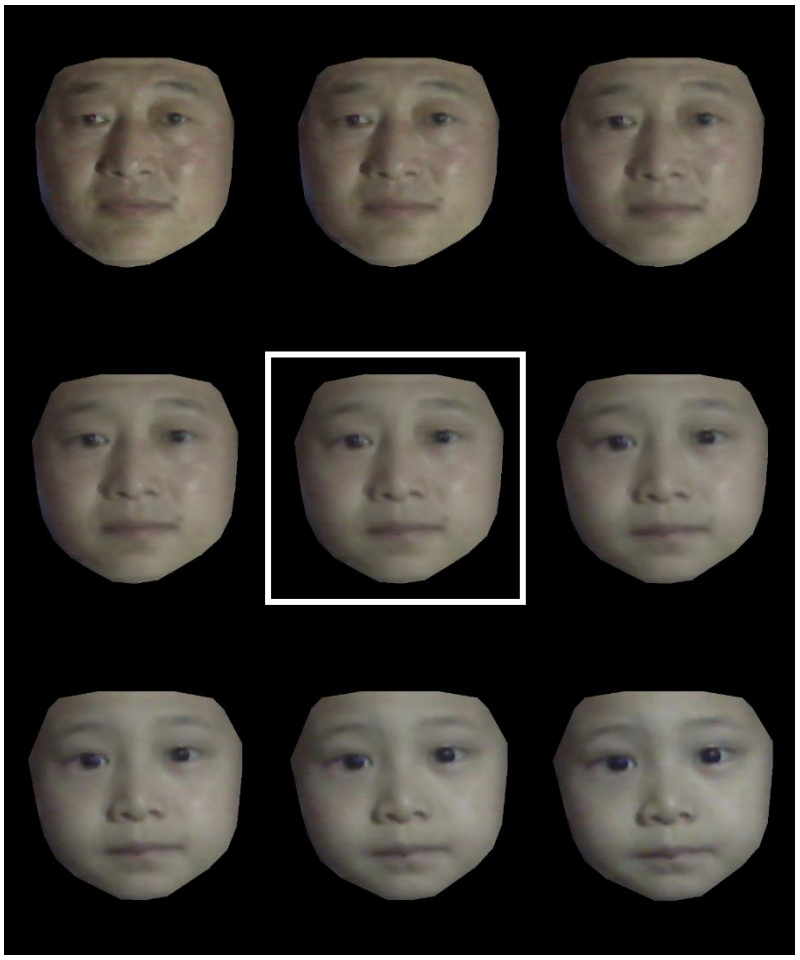
208_1.png



$\underline{\boldsymbol{M5 = Male\ To\ Child}}$

247_0.png

208_1.png



להלן התוצאות:

לפי ערך הסף הדיפולטיבי (המובנה בספריית $DeepFace$) של המודל $\boldsymbol{VGG-Face}$, שערכו הוא $\mathbf{0.68}$, התקבל כי :

Morph ID	Type A	Type B	Distance (A) from M#	Distance (B) from M#	Verified A	Verified B	Attack Successful?
M1 = 079_0 + 207_0	Male	Woman	0.459	0.671	Yes	Yes	Yes
M2 = 001_1 + 149_0	Male	Male	0.319	0.536	Yes	Yes	Yes
M2 = 136_1 + 063_1	Woman	Woman	0.389	0.511	Yes	Yes	Yes
M4 = 318_1 + 208_1	Woman	Child	0.423	0.464	Yes	Yes	Yes
M5 = 247_0 + 208_1	Male	Child	0.493	0.584	Yes	Yes	Yes

לפי ערך הסף האופטימלי לפי הדטה סט שלי (~ 1000 הזוגות הספצייפים שנבחרו) שנמצא בחלק א,
של המודל $\boldsymbol{VGG-Face}$ שערכו הוא $\mathbf{0.541}$, התקבל כי :

Morph ID	Type A	Type B	Distance (A) from M#	Distance (B) from M#	Verified A	Verified B	Attack Successful?
M1 = 079_0 + 207_0	Male	Woman	0.459	0.671	Yes	No	No
M2 = 001_1 + 149_0	Male	Male	0.319	0.536	Yes	Yes	Yes
M2 = 136_1 + 063_1	Woman	Woman	0.389	0.511	Yes	Yes	Yes
M4 = 318_1 + 208_1	Woman	Child	0.423	0.464	Yes	Yes	Yes
M5 = 247_0 + 208_1	Male	Child	0.493	0.584	Yes	No	No

בידוע, מטרת הניסוי הייתה לבדוק האם ניתן לייצר תמונת מורף שמצליחה לעבור אימות כשני אנשים שונים.
הצלחה מוגדרת כמצב שבו המרחקים (כפי שהוסבר בחלק א- $cosine\ distance$) של המורף מהתמונות המקוריות נמוכים מהסף, והמודל מאשר את ההתאמה ($verified = True$) לשני הצדדים.

מהטבלאות שלהלן, ניתן להסיק כי :
עבור הסף הדיפולטיבי שערכו הוא 0.68 , קיימות 5 מתוך 5 הצלחות, כלומר, כל תמונות המורף הצליחו לעבור אימות כשני הצדדים.
עבור הסף המחמיר יותר = האופטימלי מחלק א, שערכו הוא 0.541, קיימות 3 הצלחות מתוך 5.
הפער בין ערכי הסף מדגיש את ההשפעה של רמת הרגישות של המודל. כאשר הסף גבוה (0.68), המודל נוטה לאשר התאמות גם עבור תמונות מורף , מה שמעיד על פגיעות למתקפות. לעומת זאת, הורדת הסף ל־0.541 מחזקת את ההגנה ומפחיתה את אחוזי ההצלחה של ה $Morphing\ attack$, אך עדיין מאפשרת הצלחה במספר מקרים, מה שמעיד על כך שגם סף זה אינו חסין לחלוטין.

ניתן לראות שההתקפות נטו להצליח כאשר המרחקים של תמונת המורף מהתמונות המקוריות היו קרובים מאוד לסף, ולעיתים אף לשניהם. כאשר אחד מהמרחקים היה גבוה מהסף, התבצעה דחייה באימות.
לדוגמה: במקרה של $M2 = 001_1 + 149_0$ המרחקים לשני המקורות היו נמוכים באופן משמעותי מהסף (0.319 ו־0.536), ולכן המורף עבר אימות מוצלח אצל שני הצדדים , גם עבור המקרה של הסף המחמיר.

הניסוי מראה כי המודל $VGG-Face$ פגיע למתקפות מורפינג כאשר משתמשים בסף מקל (0.68), אך הפגיעות מצטמצמת כאשר עובדים עם סף מחמיר המבוסס על עקומת ROC . למרות זאת, הצלחות גם תחת סף 0.541 מדגימות שלמרות השיפור בהגנה, עדיין קיימת רגישות, במיוחד כאשר המורף דומה מאוד לשני המקורות.

מידע על הספריות בהן השתמשתי בשני חלקי הפרויקט :

ספריה	גרסה	מקור להורדה
deepface	0.0.93	https://github.com/serengil/deepface
opencv-contrib-python	4.5.5.64	https://pypi.org/project/opencv-contrib-python
tensorflow	2.19.0	https://pypi.org/project/tensorflow
pandas	2.3.0	https://pypi.org/project/pandas
matplotlib	3.10.3	https://pypi.org/project/matplotlib
pillow	11.3.0	https://pypi.org/project/Pillow
imageio	2.37.0	https://pypi.org/project/imageio
mtcnn	1.0.0	https://pypi.org/project/mtcnn
retina-face	0.0.17	https://pypi.org/project/retina-face

קבצי הקוד של חלק א' :

$generate_pairs.py$ - יוצר זוגות תמונות: "אמיתיים" (מאותו אדם) ו"מתחזים" (מאנשים שונים), ושומר אותם לקובץ CSV לשימוש בהשוואות. (קוד זה יצר את הקובץ $pairs.csv$ אשר מכיל 1000 זוגות)

$run_deepface.py$ - מריץ השוואות בין זוגות תמונות פנים באמצעות אחד המודלים של $DeepFace$ שהם $FaceNet$ ו- $VGG-Face$ ומפיק ציון דמיון לכל זוג. (קוד זה יצר את קבצי ה $scores_X.csv$ של מודלים אלו)

$run_opencv.py$ - משווה בין זוגות תמונות פנים באמצעות המודל $LBPH$ של $OpenCV$ ומחשב את המרחק בין התמונות לפי הזיהוי. (קוד זה יצר את קבצי ה $scores_OpenCV.csv$)

$evaluate_thresholds.py$ - מחשב את ערכי FMR ו־ $FNMR$ עבור טווח של ספים: $thresholds$, לפי קובץ תוצאות הניקוד, עבור כל אחד מן הכלים/מודלים של זיהוי פנים. (קוד זה יצר את קבצי ה $thresholds_X.csv$)

$find_optimal_threshold.py$ - בודק עבור כל כלי זיהוי פנים מהו ערך הסף האופטימלי שממזער את סכום השגיאות $(FMR + FNMR)$ ושומר את הערכים בטבלה מרוכזת. (קוד זה יצר את הקובץ $optimal_threshol_X.csv$)

$analyze_predictions.py$ - מנתח את תוצאות ההשוואה של מודל זיהוי פנים נתון, על סמך סף אופטימלי שנמצא מראש ומחשב את רכיביה של $confusion\ matrix$ (שהם TP, TN, FP, FN) ואת אחוזי הדיוק והשגיאה. (קוד זה יצר את קבצי ה $prediction_analysis_X.csv$ של מודלים אלו)

$plot_roc.py$ - יוצר גרף ROC בודד עבור כלי זיהוי פנים ספציפי לפי קובץ ספים $(thresholds_X.csv)$ שהופק מראש. (קוד זה יצר את קבצי ה $roc_X.png$)

$plot_multi_roc.py$ - יוצר גרף ROC מרוכז המציג את ביצועי שלושת המודלים להשוואה ויזואלית בין הכלים. (קוד זה יצר את הקובץ $roc_multi_curve.png$)

קבצי הקוד של חלק ב' :

$morph_faces_local.py$ - מפעיל את אלגוריתם המורפינג על שתי תמונות פנים שנבחרו, ומייצר את רצף הפריימים של תהליך המורפינג בעזרת $facemorpher + dlib$. (קוד זה יצר את התיקיות של הדוגמאות של הזוגות $frames_X_T_Y + frame\#\#\#.png$)

$generate_morph_outputs.py$ - יוצר גריד של תמונות מורפינג (3×3) ו־GIF מונפש מתוך רצף פריימים של תהליך המיזוג בין שתי תמונות מקור. (קוד זה יצר את קבצי ה $morph_animation_9frames.gif + morph_grid_3x3.png$)

$analyze_morph_with_deepface.py$ - סקריפט שמשווה בין תמונת מורף לבין שתי תמונות המקור שלה באמצעות מודל VGG-Face, ובודק האם היא מצליחה לעבור אימות עבור כל אחד מהם. (קוד זה יצר את קבצי ה $vggface_morph_attack_results_X_T_Y.json$)