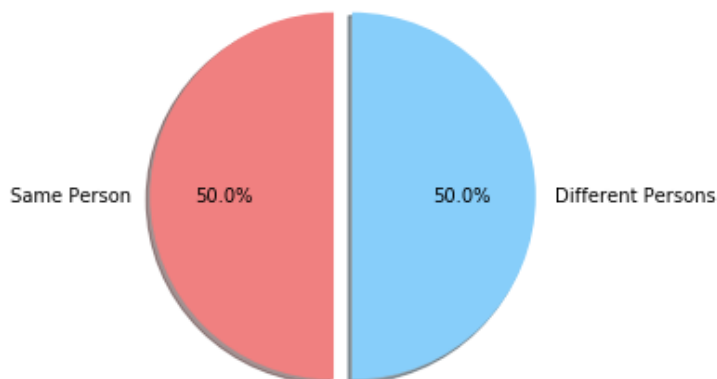


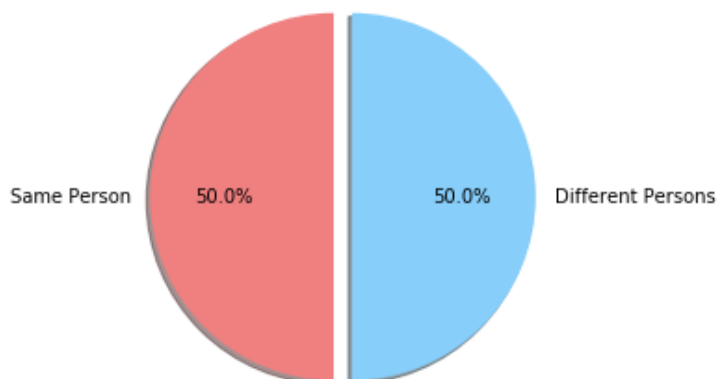
מבוא ללמידה עמיקה – דו"ח תרגיל 2

Exploratory Data Analysis

Train Set ###
Number of examples: 2200



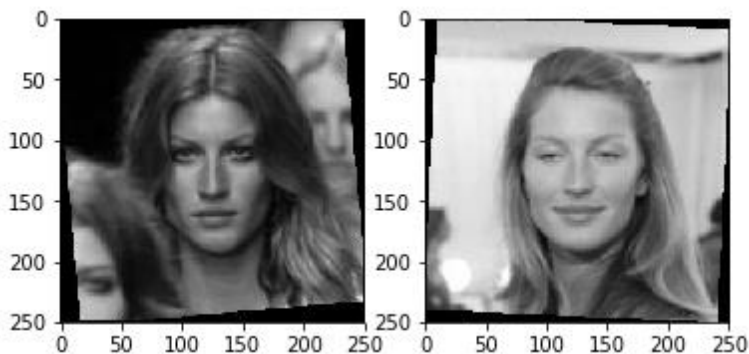
Test Set ###
Number of examples: 1000



Number of unique people in train and test sets: 5749

דוגמא לזוג תמונות מסט נתוני האימון, והסיווג האמיתי שלהם:

Label is: 1 - Same Person



לאורך התרגיל עשינו מספר ניסויים עם ארכיטקטורות והיפר-פרמטרים שונים, אותם נתאר בכל שלב.

בכל הניסיונות, השתמשנו בקריטריון עצירה עם הפרמטרים $\text{min_delta}=0.02$, $\text{patience}=5$ שמנער אחר המטריקה של val_loss . בנוסף, בגלל בעיות זכרון ו-GPU החלטנו לממש generator שמייבא לזכרון את התמונות בכל batch בנפרד, מותאם לשימוש ב-keras בתהליך האימון והולידציה.

בנוסף, לאחר מספר ניסויים, החלטנו לעשות "cropping" לתמונות ולקחת את מרכז התמונה, כך שגודל כל תמונה ירד מ- 250×250 לגודל של 150×150 . מספר הערוצים משתנה בין הארכיטקטורות בהתאם.

בנוגע לאסטרטגיית הולידציה שלנו, החלטנו לקחת את 600 האנשים הראשונים שמופיעים בנתונים הכלליים (ולא בטסט), מתוכם קיימים רק 307 דוגמאות שהאנשים מופיעים בהן, ומתוכם רק 87 דוגמאות עם סיווג 1. לכן, הוצאנו את כל 307 הדוגמאות של האנשים האלו, וביצענו under-sampling ע"י מחיקה של דוגמאות עם סיווג 0 עד לשוויון בהתפלגות הסיווגים. לבסוף, נשארו עם 174 דוגמאות לסט הולידציה, בהן השתמשנו בתהליך האימון עבור early stopping ושערוך של ביצועי המודל. אסטרטגיה זו לא תמיד הייתה טובה, לכן בחלק מהניסויים ויתרנו לחלוטין על סט ולידציה, לאחר מספר ניסיונות עם אסטרטגיות שונות.

ארכיטקטורה 1 – Koch et al.

מימשנו תחילה את הארכיטקטורה כפי שכתוב במאמר, שזה אומר רשת סיאמית שמורכבת מ-2 רשתות "embedding" שכל אחת מורכבת מ-4 שכבות קונבולוציה (פילטרים: $64 \rightarrow 128 \rightarrow 128 \rightarrow 256$) ואז שכבת Dense בעלת 512 ניוונים. שכבות הקונבולוציה בעלות אקטיביציית ReLU, וה-Dense בעלת tanh. לאחר 2 הרשתות הנ"ל, קיימת שכבה שמחסרת בין 2 הפלטים בערך מוחלט, ולבסוף שכבה עם נורון בודד ואקטיביצייה של sigmoid על מנת לחזות את הסתברות השיוך הבינארי. המודל משתמש בפונקציית הפסד של binary cross-entropy ואופטימיזר של Adam עם $\text{learning rate} = 0.00006$.

אף התאמנו את האתחולים של המשקולים השונים (w, b) כפי שהיה במאמר בעזרת פונקציות מתאימות ל-Keras.

סיכום של רשת אחת מתוך כל הרשת הסיאמית לדוגמא:

Layer (type)	Output Shape	Param #
conv2d_9 (Conv2D)	(None, 141, 141, 64)	6464
max_pooling2d_7 (MaxPooling2D)	(None, 70, 70, 64)	0
conv2d_10 (Conv2D)	(None, 64, 64, 128)	401536
max_pooling2d_8 (MaxPooling2D)	(None, 32, 32, 128)	0
conv2d_11 (Conv2D)	(None, 29, 29, 128)	262272
max_pooling2d_9 (MaxPooling2D)	(None, 14, 14, 128)	0
conv2d_12 (Conv2D)	(None, 11, 11, 256)	524544
flatten_3 (Flatten)	(None, 30976)	0
dense_4 (Dense)	(None, 512)	15860224
Total params: 17,055,040		
Trainable params: 17,055,040		
Non-trainable params: 0		

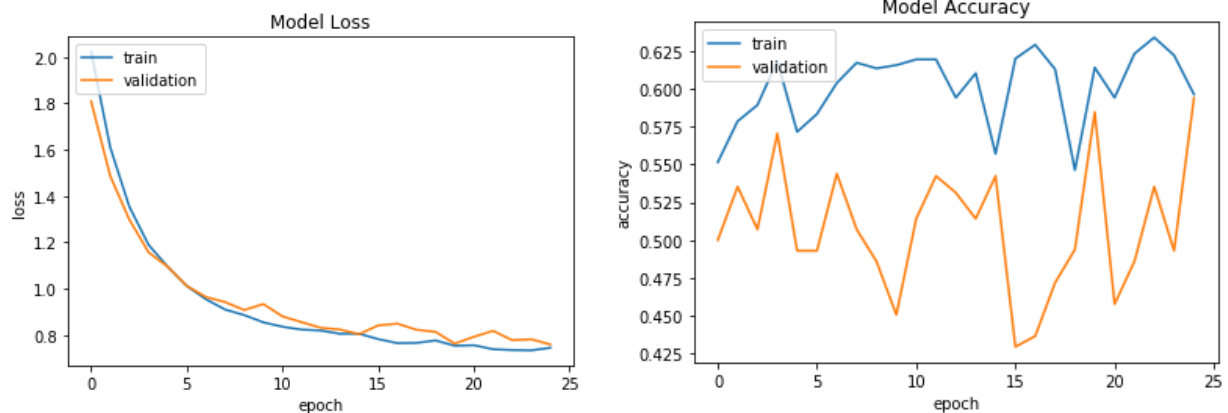
סיכום תהליך האימון:

Training process of 24 epochs finished in 227.44 seconds

Train Set -- Loss: 0.76 -- Accuracy: 0.57

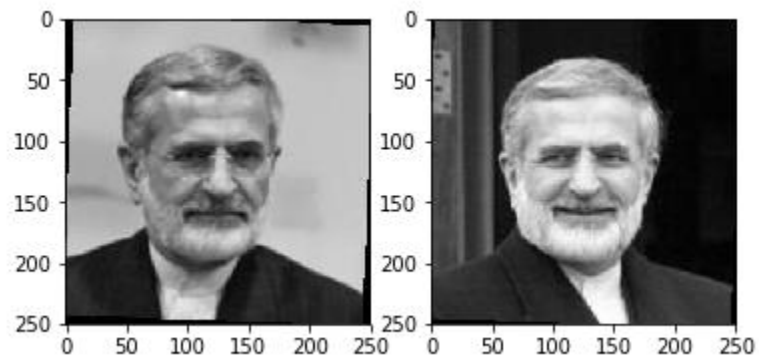
Validation Set -- Loss: 0.76 -- Accuracy: 0.54

Test Set -- Loss: 0.73 -- Accuracy: 0.63



ניתן לראות לפי גרף ה-loss שאנחנו לא נמצאים במצב של overfitting, למעט epochs מעטים לאורך התהליך.

דוגמא לסיווג לא נכון של המודל:



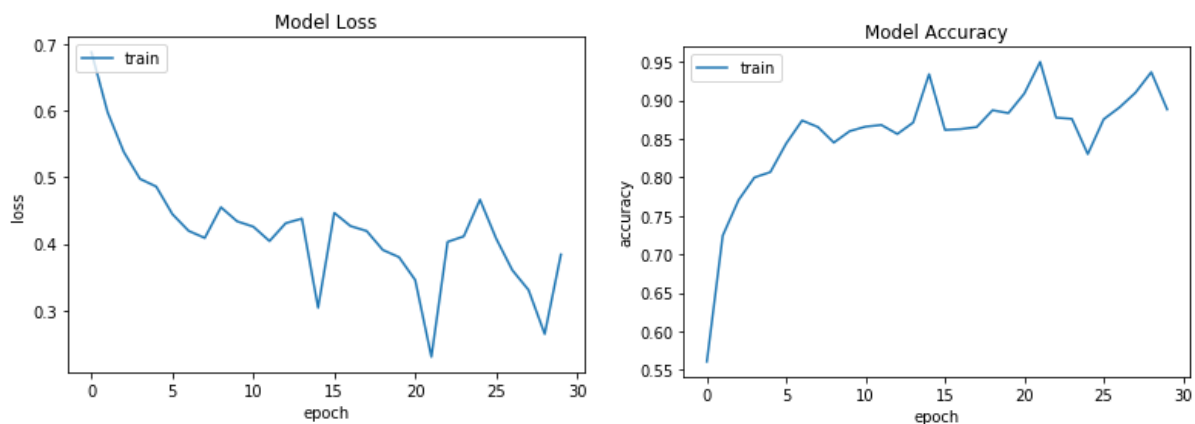
אנו מניחים שהסיווג הלא נכון קרה בגלל רקע השונה בין התמונות, למרות שזה אותו האיש ב-2 התמונות. רקע שחור כמו בתמונה הימנית כנראה שונה מדי מהרקע האפורי בתמונה השמאלית, והרשת לא למדה להתעלם מהרקעים בתמונות. אפילו החליפה שהאיש לובש היא זהה. עוד סיבה יכולה להיות המשקפיים – בתמונה הימנית האיש לא לובש משקפיים.

VGG16 – 2 ארכטיקטורה

כאן במקום להשתמש ברשת קונבולוציות מהמאמר כ"רגל" ברשת הסיאמית הכוללת, השתמשנו ברשת מוכנה מ-VGG16 – Keras עם משקולות שהתאמנו על סט הנתונים imagenet. החלטנו לעצור את אימון המשקולות ב-5 השכבות האחרונות ברשת זו, ולתת לשאר השכבות ברשת הסיאמית להתאמן כרגיל. שאר השכבות ברשת הסיאמית נשארו כפי שהזכרנו במודל הראשון.

בנוסף, בניסוי זה החלטנו לוותר על סט ולידציה ולאמן את הרשת במשך 30 epochs מלאים ללא early_stopping. אכן השגנו תוצאות מעט טובות יותר על סט הבדיקה, מאשר הניסוי הראשון.

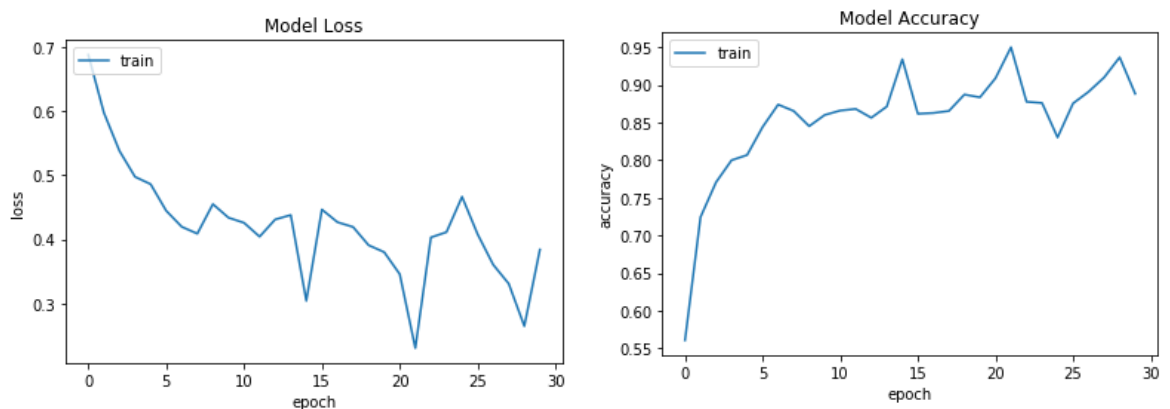
```
Train Set -- Loss: 1.11 -- Accuracy: 0.59  
Validation Set -- Loss: 1.05 -- Accuracy: 0.62  
Test Set -- Loss: 0.64 -- Accuracy: 0.68
```



ארכיטקטורה 3 – VGGFace [VGGFace University of Oxford]

מכיוון שבמודל הקודם לקחנו משקולות שאומנו על תמונות של אובייקטים, רצינו לנסות לקחת משקולות שאומנו על פרצופים של בני אדם, בתקווה שזה יעבוד טוב יותר על הנתונים שלנו. גם הארכיטקטורה של VGG16 בה השתמשנו פה מעט שונה מהדוגמא הקודמת, מכיוון שהיא מותאמת לנתונים של VGGFace. ניתן לראות את הארכיטקטורה המלאה בקוד עצמו.

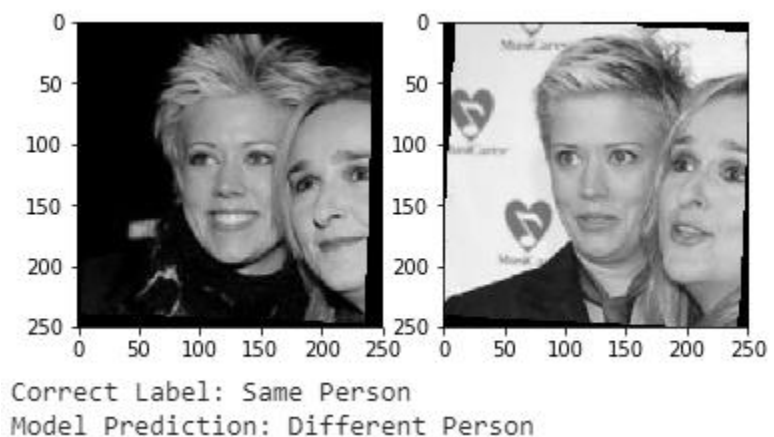
כאן גם העדפנו להעלות את מקדם הלמידה באופטימיזר של Adam ל-0.001. בנוסף, בגלל שיש יותר פרמטרים כאן וזמן הלמידה ארוך מדי, החלטנו גם להגביל את סיבובי הלמידה ל-10 epochs. לעומת הניסויים הקודמים, כאן לא השתמשנו ב-cropping וגם שכפלנו את הערוצים בתמונה ל-3 ערוצים זהים, כל זאת בשביל להתאים את הנתונים לרשת הייחודית.



Train Set -- Loss: 0.61 -- Accuracy: 0.73
Test Set -- Loss: 0.68 -- Accuracy: 0.58

ניתן לראות שהשגנו תוצאות פחות טובות מהניסויים הקודמים, אולי בגלל שהקפאנו את כל השכבות מהרשת של VGGFace ורק השכבה האחרונה ברשת הסיאמית למדה, או בגלל שלמדנו רק לאורך 10 epochs.

דוגמא לסיווג לא נכון:



כאן אנו מניחים שזה אכן בגלל הרקע השונה, או אולי בגלל התספורת של האישה משמאל.

סיכום

ראינו שהתוצאות הטובות ביותר על סט הבדיקה הוא במודל הראשוני, וגם שם אחוז הדיוק לא גבוה במיוחד. לדעתנו, בגלל הגודל הקטן של סט הנתונים, והתמונות הלא מגוונות, שהן גם ככה ב-grayscale, הרשת הסיאמית לא יכולה ללמוד יותר מדי. כמחשבה לעתיד, נרצה לקחת רשת נוירונים כדוגמאת VGG16, להשתמש בה כ-feature extractor כפי שעשינו ולהכניס אותה למודל ML קלאסי כמו SVM Regressor – ייתכן שכך נקבל ביצועים גבוהים יותר ממה שהשגנו.