האוניברסיטה הפתוחה

המחלקה למתמטיקה ולמדעי המחשב

שיטות לזיווג תמונות נטולות אילוצים של פנים

עבודה מסכמת זו הוגשה כחלק מהדרישות לקבלת תואר

"מוסמך למדעים" .M.Sc במדעי המחשב

באוניברסיטה הפתוחה

החטיבה למדעי המחשב

על-ידי

טל אגמון

העבודה הוכנה בהדרכתו של ד"ר טל הסנר

2011 מרץ

תוכן העניינים

1 רשימת איורים

2 רשימת טבלאות

3 תקציר

זיהוי פנים, מטלה כמעט מובנת מאליה אצל בני אדם, היא מטלה קשה ומורכבת מאוד עבור מחשב. זאת משום שייצוג של תמונה דיגיטלית הוא מטריצה של מספרים (פיקסלים). שינויים של זווית של הפנים, תאורה, אביזרים כמו משקפיים, שפם או זקן, שינויים בהבעות הפנים וכדומה גורמים לשינויים גדולים בערכי הפיקסלים של תמונה.

בעיית זיהוי פנים בתמונה מעוררת עניין רב בשנים האחרונות הן בעולם האקדמי והן בתעשייה. בעבודה זו נעסוק בתת בעיה שלה שהיא בעיית זיווג תמונות נטולות אילוצים של פנים (face-matching in unconstrained images). בבעיה זו הקלט הוא זוג תמונות של פני אנשים לא מוכרים, והפלט הוא תשובה חיובית אם בתמונות מופיעים פנים של אותו אדם, או תשובה שלילית אם הפנים הם של אנשים שונים. התמונות הן נטולות אילוצים מאחר ואין הגבלה על התמונות מבחינת זווית של הפנים, תאורה, מיקוד, רזולוציה, הבעות פנים משקפיים וכו', בפרט, האדם המצולם אינו משתף פעולה עם הצלם ולאו דווקא מודע לכך שהוא מצולם.

טכנולוגיות של זיהוי פנים (Face Recognition Technology - FRT) מצויות היום במאגרים ביומטריים, במערכות ביטחוניות ובמאגרי תמונות באינטרנט (למשל מנועי חיפוש ורשתות חברתיות). יחד עם זאת, תחום זיהוי הפנים מעורר עניין בעיקר ב-20 השנים האחרונות, רק לאחרונה פותחו שיטות המציגות אחוזי דיוק גבוהים יחסית ליכולת אנושית בנתונים מסוימים ועדיין קיים צורך לשפר את ביצועיהם, במיוחד בנתוני סביבה קשים. לכן, ניתן לחזות שבעתיד נראה עליה משמעותית בכמות השימושים בטכנולוגיה זו.

עבודה זו סוקרת מספר שיטות לפיתרונה של בעיית זיווג תמונות נטולות אילוצים של פנים. שיטות אלה מציגות תוצאות על מאגר הנתונים LFW – Labeled Faces in the Wild [1], מאגר נתונים מקובל לבדיקת ביצועים של זיווג פנים. מאגר זה מכיל תמונות שנאספו מהאינטרנט ועל כן מכילות גיוון בתכונות כמו תאורה, זוויות פנים, הבעות וכו' הדומה באופן יחסי לגיוון הקיים במציאות. שיטות אלה הן שיטות חדישות וביצועיהן הן מבין הטובים ביותר שפורסמו עד היום עבור בעיה זו.

4 מבוא

בבעית זיהוי פנים (face recognition) המטרה היא להצביע על זהותו של אדם אחד או יותר מתוך תמונה או מתוך וידיאו. בבעיית זיהוי פנים התשובה הרצויה היא זיהוי של אדם ספציפי (למשל שם או מספר זיהוי), זאת בניגוד לבעיית איתור פנים (face detection) שבה מוצאים את המיקומים והגדלים (אם ישנם כאלה) של הפנים בתמונה. השלב הראשון בבעיית זיהוי פנים הוא איתור פנים וברוב השיטות לזיהוי פנים תמונות הקלט הן תמונות שעברו תהליך ראשוני של איתור פנים [2].

ניתן לחלק את קבוצת הבעיות של זיהוי פנים לשלושה סוגים:

1. מציאת זהות של פנים (face identification) - בבעיה זו נתונה תמונה ויש להצביע על זהותו של האדם בתמונה מתוך מאגר של זהויות נתונות מוכרות. לשם כך מתבצע מעבר על המאגר המוצא את האדם במאגר שהכי דומה לתמונה הנתונה. דוגמא לבעיה כזאת היא מציאת זהות של אדם שצולם במצלמת מעקב מתוך מאגר של עבריינים.
2. אימות זהות (face verification) – בהינתן מודל פנים בעל זהות ידועה, יש לאשר או להפריך זהות זו בתמונה. דוגמא לבעיה כזאת היא למשל מציאת תמונות של משתמש כלשהו בפייסבוק במאגר תמונות של חבריו. (זהות המשתמש נלמדה קודם לכן מתוך התמונות המופיעות בפרופיל שלו. עבור כל תמונה של חבר, יש לאשר או להפריך את זהותו של המשתמש בתמונה).
3. זיווג פנים לא ידועות (pair-matching) – בהינתן שתי תמונות של אנשים אשר מעולם לא הוצגו בפני המערכת, יש לקבוע האם אותו אדם מצולם בשתיהם. דוגמא לבעיה זו היא זיהוי אדם לפי תעודה כלשהי. בעיה זו שונה מבעיית אימות זהות בכך ששתי התמונות הנתונות אינן מוכרות והמערכת איננה יכלה להתבסס על הכרות מוקדמת עם תמונות קודמות של אותם פנים המופיעות באחת מהן. בבעיית אימות זהות, לעומת זאת, לפחות תמונה אחת נתונה מראש בתהליך הלמידה.

בעית זיהוי פנים מעוררת עניין לא רק בקרב חוקרים של מדעי המחשב אלא גם בקרב נוירוביולוגים ופסיכולוגים העוסקים בהבנה של תהליכי הראייה אצל בני אדם. שיתוף פעולה בין דיסיפלינות מצד אחד יכול להביא לשיפור בשיטות לראייה ממוחשבת המשתמשות במנגנון הראייה האנושי כהשראה ומצד שני הישגים בתחום הראייה הממוחשבת יכולים לשפוך אור על הבנת תהליכי ראייה אנושית. כיום, שיטות לזיהוי פנים, כמו הרבה שיטות נוספות מתחום הראייה הממוחשבת, משתמשות בטכניקות של למידה חישובית אשר התפתחו בהשראת מנגנון הלמידה האנושי.

4.1 מבוא ללמידה וראייה חישובית

ראייה חישובית (computer vision) עוסקת בפיתוח כלים שמאפשרים לחקות בכמה שיותר דיוק ניתוחים ועיבודי נתונים שבני אדם מבצעים על תמונות. ניתוחים אלה הם למשל זיהוי ואימות של אנשים בתמונה, זיהוי של אותיות בכתב יד, איתור וסיווג של פנים ואובייקטים שונים בתמונה, סיווג סוגים של תמונות, ביצוע מניפולציות שונות על תמונות ועוד. אלגוריתמים של ראייה חישובית משתמשים היום לרוב בשיטות של למידה חישובית.

בשיטות של למידה חישובית מפעילים תהליך של למידה מתוך קבוצה נתונה של דוגמאות הנקראת קבוצת למידה (training set). מתהליך למידה זה נוצר אלגוריתם לפתרון הבעיה. קבוצת הנתונים אשר בזמן הבחינה, מהווה קלט לאלגוריתם שנוצר, נקראת קבוצת הבחינה (test set). משתמשים בדרך כלל בלמידה חישובית על-מנת לפתור בעיות אשר לא ידועים עבורם פתרונות מסורתיים, בדרך כלל משום שאוסף של כל הקלטים וההתנהגויות האפשריות שלהן הוא מורכב מדי וגדול מדי ועל-כן דרך היחידה הידועה להגדיר את התנהגות האלגוריתם היא על-פי דוגמאות.

למידה חישובית יכולה להיות מונחית (supervised) או בלתי מונחית (unsupervised). בלמידה מונחית הקלט עבור תהליך הלמידה הוא קבוצת של דוגמאות כאשר עבור כל דוגמא נתון הפלט המבוקש. בלמידה שאינה מונחית, לעומת זאת, לא יודעים בתהליך הלמידה את הפלט עבור קבוצת הדוגמאות הנתונה.

4.2 היסטוריה של זיהוי פנים

עד סוף שנות ה-80, התבססו טכנולוגיות של זיהוי פנים על שיטות סטטיסטיות ושיטות גיאומטריות פשוטות (למשל מדידת מרחקים בין איברים של הפנים) שלא היו לגמרי אוטומטיות משום שהם הצריכו מהמפעיל לאתר את חלקי הפנים השונים ידנית.

בשנת 1988, השתמשו קירבי וסירוביץ' (Kirby and Sirovich) בטכניקה הנקראת ניתוח גורמים ראשיים (PCA – Principle Component Analysis [3]). לייצוג של תמונות פנים ע"י מרחב מצומצם שהוקטורים העצמיים (ראה 14.1.5) שלו נקראים פנים עצמיים (eigenfaces) [4]. שיטה זו מהווה אבן-דרך בהיסטוריה של זיהוי פנים מאחר והיא מצליחה לייצג תמונה של פנים באמצעות פחות ממאה ערכים [3], יכולה לשמש עבור איתור פנים [5], וב-1991 השתמשו בה לראשונה טורק ופנטלנד עבור זיהוי פנים אוטומטי [6]. שיטה זו הגבירה את העניין בפיתוח של מערכות אוטומטיות לזיהוי פנים והרבה שיטות שפותחו מאז מבוססות עליה ומרחיבות אותה.

בשנות ה-90 טכנולוגיות זיהוי פנים החלו לעורר עניין נרחב והחלו לאמץ כלים שונים מתחומים כמו רשתות נוירונים ([7],[8]), זיהוי תבניות ([9],[10]) ולמידה חישובית ([11],[5]). השימוש בלמידה חישובית שעד היום מהווה כלי מרכזי בזיהוי פנים מצריך מאגר נתונים לתהליך הלמידה. ככל שמאגר נתונים זה מכיל יותר תמונות פנים עבור כל אדם, כך הזיהוי יכול להיות מדויק יותר.

מצד שני שימוש במאגר פנים גדול מדי הוא בעייתי, מאחר שלרוב קשה לאסוף הרבה תמונות פנים, האחסון דורש הרבה מקום, נפח הנתונים מאריך את זמן הריצה ולפעמים גם בנסיבות מסוימות פשוט אין בנמצא הרבה תמונות פנים עבור כל אדם או שהן זמינות רק בתהליך הבחינה (למשל במערכת לזיהוי אדם מדרכון, יש רק תמונה אחת לכל אדם).

בשל נסיבות אלה, התפתחה הבדלה בין שתי בעיות שונות [12]:

1. בעיה מרובה דוגמאות (multiple samples problem) לדוגמא – [4],[13].
2. בעיה עם דוגמא יחידה (one-sample problem) לדוגמא – [14],[15].

בנוסף, מאגר נתונים גדול המכיל תמונות פנים של הרבה אנשים, הוא בעייתי משום שהסיכוי לזיהוי שגוי גובר בגלל הריבוי בתמונות דומות של אנשים שונים. לכן ישנו יתרון למאגר נתונים מהטיפוס בו משתמשים בבעיית זיווג פנים, סוג שונה של בעיית זיהוי פנים. הקלט הוא שתי תמונות נטולות אילוצים (ללא מגבלות לגבי תאורה, זוית פנים, הבעות וכדומה) שלא נראו בעבר, והמטרה היא לאמת או להפריך את העובדה כי שתי תמונות אלה מכילות פנים של אותו אדם. מאגר נתונים עבור בעיה זו מכיל זוגות של פנים חיוביים (של אותו אדם) או שליליים (של אנשים שונים) ובמקום ללמד על מודלים של פנים בעלי זהות ידועה, מנסים ללמוד מהו דימיון בין שתי תמונות, מתי ניתן לומר ששתי תמונות הן תמונות של אותו אדם.

נשים לב שבעיה זו היא אינה תת-בעיה של הבעיה עם דוגמא יחידה משום שבבעיה זו, בניגוד לבעית הדוגמא היחידה, אין דוגמא יחידה הנתונה בזמן תהליך הלמידה אלא שתי התמונות נתונות בזמן הבחינה. הלמידה מתבצעת על מאגר של זוגות של תמונות שכולן שונות מהתמונות המשמשות בזמן הריצה כקלט. בנוסף, הפלט הוא תשובה חיובית או שלילית ולא בחירה של זהות מתוך מאגר דוגמאות.

4.3 חשיבותה של הבעיה

בעיית זיהוי פנים מעוררת עניין רב הן בעולם האקדמי והן בתעשייה וקיימים מספר גורמים לכך: צמיחה של מערכות ממוחשבות המצריכות זיהוי ממוחשב (שאחרת מתבצע באמצעות סיסמא), צמיחה של מאגרי תמונות (למשל ברשתות חברתיות), העובדה שבני אדם באופן טבעי משתמשים בזיהוי פנים ככלי העיקרי לזיהוי אנשים, רמת הדיוק ההולכת וגדלה שאליה ניתן להגיע באמצעות זיהוי פנים והיכולת של מחשב, בניגוד לבני אדם, לזכור מספר רב של פנים של אנשים. בנוסף לכך רמת ההתערבות הנמוכה הנדרשת מן האדם המזוהה גורמת לכך שזיהוי פרצופים היא טכניקה המתאימה מאוד למערכות ביטחון (ניתן לצלם ממצלמה מרוחקת ללא ידיעת האדם שאותו מזהים). רוב הציפיות ממערכות לזיהוי פנים היום הן יהוו כלי משמעותי בהגנה וביטחון ומאז אירועי ה-11 בספטמבר בארצות הברית החלה הממשלה האמריקאית לעודד ולהשקיע תקציבים במחקרים של שיטות לזיהוי פנים [16].

כל אלה הופכים זיהוי פנים ממוחשב להיות מאוד מבוקש ועל כן יש ביקוש רב לפיתוח של שיטות מהירות, מדויקות ואשר מצריכות מאגר נתונים כמה שיותר קטן. העניין הרב סביב פיתוח

מערכות שמפעילות זיהוי ממוחשב לצורכי ביטחון, נתינת הרשאות או שמירה על פרטיות, מצריך מציאת פתרונות מוצלחים לבעיית הזיווג של תמונות של פנים. כמו-כן, על-מנת להתמודד עם תנאי היום-יום של הצילום במערכות אלו, יש להצליח לעבד תמונות אשר נלקחו בסביבה שתנאיה אינם ידועים. בנוסף לכך, מאחר ובמערכות כאלה קיים נזק ממשי אם הזיהוי הוא שגוי, יש לדרוש תשובה חד משמעית ומדויקת לשאלה האם אותו אדם מופיע בתמונה. . פיתוח שיטות של זיווג פנים יכול לסייע גם עבור פיתוח של מערכות זיהוי פנים אחרות, מאחור הוא חוקר איך ניתן לקבוע דימיון בין תמונות פנים. שיטות מסוימות של זיווג פנים יכולות אף לעזור בבעיות זיהוי שונות [17].

4.4 מדוע זוהי בעיה קשה?

זיהוי פנים, מטלה כמעט מובנת מאליה אצל בני אדם, הופכת להיות קשה במיוחד עבור מחשב משום שתמונה דיגיטלית היא בעצם מטריצה של מספרים (פיקסלים). שינויים של זווית של הפנים, תאורה, אביזרים כמו משקפיים, שפם או זקן, שינויים בהבעות הפנים וכדומה יכולים לגרום לתמונה להיות שונה מאוד (מבחינת ערכי הפיקסלים) מתמונה אחרת של אותו אדם. למשל, בהרבה מקרים, תמונות של אותו אדם בתאורה שונה שונות יותר מבחינה מספרית מאשר תמונות של אנשים שונים בתאורה זהה.

השינויים בתמונות של אותם פנים יוצרים מגוון רחב של תמונות אפשריות עבור כל אדם (ראה איור 1) וניתן לחלקם ל-3 קבוצות ([18]):

1. שינויים פיסיים בפנים: איפור, שפם וזקן, שינויים בגיל, משקפיים, הבעות שונות וכו'.
2. שינויים גיאומטריים: שינויים בזויות של הפנים, שינויים בזויות של המצלמה.
3. שינויים הקשורים בצילום ובתמונה: שינויי תאורה, איכות המצלמה, רעש, השפעות של דחיסת התמונה, ומגבלות רזולוציה.

איור 1 – שינויים בפנים של אותו אדם

דוגמאות לשינויים בפנים של אותו אדם: שינויי זויות, תאורה, הבעות ומשקפיים.

4.5 מבנה העבודה

הפרקים הבאים מאורגנים באופן הבא. פרק 5 מבצע סריקה רוחבית של שיטות המקובלות היום לזיהוי פנים בכמה גישות שונות. פרק 6 מתעמק בהגדרה מדויקת של הבעיה ובמאגר הנתונים LFW. פרק 7 סוקר כמה אלגוריתמי חלוקה וסיווג שימושיים. פרקים 8-10 מתעמקים בשלוש

שיטות חדשניות לזיווג פנים את המציגות את התוצאות שלהם על מאגר הנתונים LFW. בפרק 11 השוואה בין התוצאות ששיטות אלה משיגות. פרק 12 מתאר אפליקציה המדגימה זיהוי פנים מוידיאו שפותחה ע״י שימוש בממשק של חברת face.com. מסקנות ודיון באפשרויות למחקרים עתידיים מובאים בפרק 13. נספח המכיל רקע מתמטי הנדרש להבנת העבודה נמצא בפרק 14.

5 סקירה של שיטות

פרק זה סוקר שיטות שונות של זיהוי פנים ומסווג אותן לסוגים שונים של גישות. ישנו מבחר רחב מאוד של שיטות לזיהוי פנים, על-כן לא ניתן להזכיר כאן את כולן, ננסה לסקור את השיטות שהן יחסית מוכרות יותר, נמצאות בשימוש נרחב או משמשות בסיס למספר גדול של שיטות.

נדון כאן ב-3 סוגים של שיטות:

1. שיטות גיאומטריות
2. שיטות המבוססות על תתי-מרחבים
3. שיטות מבוססות מיקרו-טקסטורה

5.1 שיטות גיאומטריות

השיטות הראשונות לזיהוי פנים ניסו להסתמך על התכונות הגיאומטריה של הפנים באופן ישיר.

שיטות אלה מחפשות חוקים שיתארו את הצורה של איברי הפנים. לשם כך מסתמכים לרוב על נקודות השוואה חשובות בפנים (fiducial points) אשר באופן יחסי אינן מושפעות משינויים בתמונות פנים של אותו אדם כמו הבעות שונות, תאורה שונה וכו'.

השיטה הראשונה לזיהוי פנים פותחה בשנות ה-60 ובה נדרש לסמן איברים של פנים באופן ידני בתמונות. המרחקים ביניהם היו מחושבים באופן אוטמטי ומושווים לאותם המרחקים של פנים הנמצאים במאגר הנתונים.

בשנות ה-70 פותחה שיטה המשתמשת ב-21 תכונות כמו צבע שיער ועובי השפתיים בכדי לשפר את אחוזי ההצלחה של הזיהוי [19], אך גם שיטה זו הצריכה סימונים ידניים של נקודות השוואה.

לקראת שנות ה-80 השאיפה לייעל את תהליך זיהוי הפנים ולהפוך אותו לתהליך שהוא אוטומטי לחלוטין, היא שהביאה למציאת שיטות חדשות לזיהוי פנים שהשפיעו רבות בהמשך, שיטות אלה הן שיטות המבוססות על תתי-מרחבים.

5.2 שיטות המבוססות על תתי-מרחבים

ניתן לייצג תמונה בעלת x פיקסלים כוקטור בעל x מימדים אשר איבריו הם ערכי כל הפיקסלים של התמונה. בדרך זו ניתן להתייחס לכל תמונה כוקטור במרחב רב-מימדים. מאחר ובתמונה חלק מערכי הפיקסלים הם פחות רלוונטיים לזיהוי ואף יכולים להוות רעש עבורו, משתמשים בהיטל של תמונות פנים אל מרחב מצומצם של מימדים שהם המימדים הרלוונטיים יותר לזיהוי. הזיהוי נקבע לפי תוצאת ההפעלה של פונקציית מרחק כלשהי המופעלת על היטל זה ועל נקודה אחרת במרחב המצומצם המייצגת פנים כלשהם או מחלקה של פנים.
ב-1991 השתמשו טורק ופנטלנד [6] לראשונה בשיטה המבוססת על תתי-מרחבים הנקראת ניתוח גורמים ראשיים (Principle Component Analysis - PCA) עבור זיהוי פנים ומאז הרבה שיטות

פיתחו והרחיבו אותה [20],[21],[22]. בשיטה זו מחשבים את מטריצת השונות (covariance matrix) של הוקטורים המייצגים את התמונות שבמאגר הנתונים המוקדש ללמידה (ראה 14.2.1). בשלב השני, מוצאים את הוקטורים העצמיים (eigenvectors, ראה 14.1.5) של מטריצת השונות ומסדרים וקטורים אלה בסדר הולך ויורד לפי ערכי המקדמים שלהם. סדר זה מגדיר את מידת ההשפעה שלהם על השונות של התמונות. וקטורים אלה בעצם מייצגים תמונות פנים הנקראות פנים עצמיים (eigenfaces). באמצעות בחירה של t וקטורים מתוך הוקטורים העצמיים הנ״ל (הבחירה מתבצעת לפי סדר החשיבות שהוזכר), יוצרים מרחב פנים מצומצם בעל t מימדים. אם-כן, מרחב הפנים המצומצם הוא מרחב הנוצר מוקטורי בסיס (ראה 14.1.4) המייצגים את הכיוונים בהם שונות הנתונים גדולה יותר. כשתמונת פנים מתקבלת, מחשבים את ההיטל שלה על מרחב הפנים המצומצם, ומוצאים את תמונת הפנים במאגר הלמידה שהמרחק האוקלידי בינה לבין הקלט במרחב הפנים המצומצם הוא הקטן ביותר.

בשיטת אנליזה של הבחנה לינארית (Linear Discriminant Analysis – LDA, [23], [24],[25]) מחפשים מרחב בעל מספר מימדים מצומצם, המורכב מצירוף לינארי של וקטורי בסיס נקראים fisherfaces, כך שהמרחקים בין נקודות מאותו סיווג במרחב זה יהיו כמה שיותר קטנים, ואילו המרחקים בין נקודות בעלות סיווג שונה יהיו כמה שיותר גדולים.

לשם כך משתמשים בשתי מטריצות: הראשונה היא מטריצה המתארת את המרחק של איברי כל מחלקה (פנים של אותו אדם) מהממוצע שלה, כלומר את הפיזור התוך-מחלקתי (within-class scatter matrix) השניה מתארת את המרחק של כל המחלקות מהממוצע של המחלקות, כלומר את הפיזור הבין-מחלקתי (between-class scatter matrix). השאיפה היא להביא למינימום את הפיזור התוך מחלקתי ולמקסימום את הפיזור הבין-מחלקתי. זיווג פנים מתבצע ע״י מציאת התמונה שהמרחק שלה מההיטל של תמונת הקלט הוא הקטן ביותר.

לאחרונה, מספר מחקרים הראו שברוב המקרים המרחב הנוצר מתמונות פנים הוא מרחב, או ליתר דיוק, יריעה (manifold), אשר אינו לינארי. PCA ו-LDA מבוססות על ההנחה שהמרחבים הנוצרים ע״י התמונות הם מרחבים לינאריים ועל כן עלולות אינן משתמשות במרחקים מתאימים בין הנקודות במרחב שמתאימות לתמונות. הבנה זו הובילה לחיפוש שיטות שאינן לינאריות וביניהן Isomap [26] ו- LLE [27], [28].

Isomap היא שיטה המשתמשת במרחקים גיאודזיים (geodesic distances) ולכן מתאימה יותר לשימוש במרחב שהוא לא לינארי. שיטה זו מורכבת מ-3 שלבים עיקריים: בשלב הראשון מגדירים עבור כל נקודה במרחב את קבוצת השכנים שלה (או לפי רדיוס נתון או ע״י אלגוריתם למציאת k השכנים הקרובים ביותר, kNN – ראה 7.3), בונים גרף שמחבר בין 2 נקודות בקשת שמשקלה הוא המרחק ביניהם אם״ם הן שכנות. בשלב השני מחשבים את כל המרחקים הגיאודזיים בין כל הנקודות ע״י שימוש באלגוריתם למציאת המסלול הקצר ביותר ובשלב השלישי משתמשים במטריצת המרחקים שנוצרה בשביל לשכן (embed) את הנקודות במרחב ממימד נמוך יותר. שיכון הוא העתקה של מבנה מתמטי אחד אל תוך מבנה מתמטי אחר אשר אינה משנה את תכונות המבנה המקורי.

שיטה נוספת לצמצום לא לינארי היא LLE – Locally Linear Embedding. שיטה זו מתבססת על האינטואיציה הגיאומטרית המנסה לשמור על יחסים בין נקודות שכנות במהלך מיפויים למרחב מצומצם. בשלב הראשון מוצאים עבור כל נקודה את השכנים שלה לפי אלגוריתם kNN שהוזכר קודם. מייצגים כל נקודה באמצעות קירוב שהוא סכום ממושקל של הנקודות המהוות את השכנים שלה. בשלב הסופי, יוצרים מיפוי של כל נקודה לנקודה במרחב עם מספר מימדים מצומצם, כך שהמרחק בין המיפוי של הנקודה מאותה משוואה המחושבת על המיפויים של השכנים יהיה מינימאלי.
ב-10 השנים האחרונות פותחו שיטות המשפרות סיווג וזיהוי פנים שמרחיבות ומפתחות את Isomap וLLE, למשל [29],[30],[31],[32].

5.3 שיטות מבוססות מיקרו-טקסטורה

בשיטות אלה משתמשים במאפיינים מקומיים (local features) של הפנים. כלומר, במקום לייצג את התמונה בכללותה ע״י וקטורים ממימד גבוה, משתמשים במתארים מקומיים (local descriptors) המתארים חלקי פנים של התמונה ובשיטות הרכבה שונות שלהם על מנת לאפיין את הטקסטורה של הפנים. שיטות כאלה הן בהרבה מקרים עמידות יותר לשינויים של זויות, תאורות והבעות שונות. החיסרון בשיטות אלה בדרך כלל הוא הקושי בהחלטה באילו מאפיינים להשתמש, בגודלן ובכמותן בכדי להשיג דיוק מקסימלי ושמירה על סיבוכיות שאינה גבוהה מדי.

שימוש במאפיינים של תבניות בינאריות מקומיות (local binary patterns - LBP) נוסה לראשונה ב-[33] ומאז משתמשים במאפיינים אלה בשיטות שונות. אופרטור מסוג LBP פועל על קבוצה של פיסקלים ומתייחס לפיקסל שבמרכזה כערך סף. הפיקסלים השכנים מקבלים ערך 0 או 1 לפי ערך סף זה. המאפיין הוא המספר הבינארי המתקבל משרשור הספרות החל מהפיקסל שמעל הפיקסל המרכזי ובכיוון השעון (ראה איור 2). מחלקים את תמונת הפנים לאזורים ועבור כל אזור מחשבים את כל אופרטורי ה-LBP האפשריים עבורו ומרכיבים היסטוגרמה המכילה את כל המספרים הבינאריים שחושבו. בשלב הסופי מרכיבים את כל ההיסטוגרמות שהתקבלו להיסטוגרמה אחת. בכדי להכריע אם שתי תמונות פנים שייכות לאותו אדם, משווים את ההיסטוגרמות באמצעות כלי השוואה סטטיסטיים.

כיום, קיימות הרחבות שונות להגדרה המקורית של LBP [34] ואחת מהן היא שימוש בגדלים שונים של קבוצות פיקסלים עליהן מופעל חישוב המתאר. שימוש נפוץ הוא בסביבה מעגלית.

הרחבה נוספת היא שימוש בתבניות אחידות (uniform patterns). מתאר LBP אחד הוא מספר בינארי שבו מספר המעברים מ-0 ל-1 או ההיפך אינו גדול מ-2, כאשר מסתכלים על המספר בצורה מעגלית. למשל התבניות 00000000, 00111100, 11100111 הן תבניות אחידות. ניסויים שונים שנערכו הראו שרוב התבניות הבינאריות המקומיות בתמונות הן אחידות, ולכן ניתן להשתמש רק בתבניות אחידות ולקודד את שאר התבניות כמספר בינארי יחיד. באמצעות קידוד חסכוני זה מקבלים סיבוכיות מופחתת.שיטות המשתמשות ב-LBP הן למשל [33],[34],[35],[36].

מאפיינים מקומיים שימושיים נוספים מתבססים על מסנני גבור (Gabor filters) [37]. מסננים אלה מורכבים ממכפלה של פונקציות גרעין גאוסיניות (Gaussian kernel functions) בסינוסואיד מורכב. מסנני גבור מעוררים התעניינות רבה משום שהם התגלו ככלי מאוד יעיל למציאת הגבול בין טקסטורות שונות בתמונה, ולכן אפשר לגלות באמצעותה צורות של אובייקטים (ראה איור 3). מתוצאות של הפעלת מסננים אלה, ניתן להרכיב מאפיינים מקומיים [38] המאפיינים את הצורות של חלקי הפנים השונים.

איור 2 – אופרטור LBP

דוגמא של הפעלת אופרטור LBP. האופרטור הקלאסי הוא בגודל 3X3. הערכים של הפיקסלים השכנים מקבלים 0 או אחד לפי ערך הסף שהוא הפיקסל המרכזי. המספר הבינארי המתאים מתקבל ע"י שרשור של ערכי ה-0,1 שהתקבלו החל מהפיקסל שמעל הפיקסל המרכזי ובכיוון השעון.

איור 3 – הפעלה של פילטר גבור

דוגמא לתמונה (התמונה המקורית מצד שמאל) לאחר הרכבה של פילטרי גבור [39].

שיטות לזיהוי פנים המשתמשות בפילטרי גבור הן למשל [40],[41],[42],[43]

שימוש במתארי SIFT (SIFT descriptors, SIFT - Scale Invariant Feature Transform [45],[44]), מתבסס על וקטורים המורכבים מנקודות השוואה בתמונה (נקודות כאלה מתארות אזורים בתמונה שאינם רגישים לשינויים כמו תאורה או זוית פנים). מוצאים נקודות כאלה בכמה מידות שונות (scales) של התמונה. נקודות כאלה הן בדרך כלל קצוות של אובייקטים בתמונה. עבור כל נקודה, שהיא בעצם אזור של שינויים בערכי הפיקסלים, מוצאים גם את האוריינטציה (orientation) שהיא הכיוון של הגרדיאנט (gradient, ראה 10.1.1). את אזורי נקודות ההשוואה מקודדים ומרכיבים מהם היסטוגרמות שהערכים שלהם ירכיבו את הוקטורים שהם מתארי ה-SIFT, הנקראים גם מפתחות SIFT (SIFT keys). בשלב הסופי משווים את הוקטורים המתארים באמצעות מרחקים אוקלידיים למציאת דמיון בין פנים הקובע אילו פנים

שייכים לאותו אדם. שיטה חדשה המשלבת מאפייני Gabor, LBP ו-SIFT שמציגה תוצאות טובות על מאגר הנתונים LFW היא [46].

6 הגדרת הבעיה וסביבת העבודה

הבעיה שבה נעסוק מעתה היא בעיית זיווג תמונות נטולות אילוצים של פנים. כזכור, הקלט הוא זוג תמונות (לאחר שאותרו בהם פנים) של פנים אשר לא נראו בעבר, והפלט הוא התשובה לשאלה האם אותו אדם מופיע בשתי התמונות. התמונות הן נטולות אילוצים (unconstrained). כלומר, אין כל הגבלה על התמונות מבחינת זווית של הפנים, תאורה, מיקוד, רזולוציה, הבעות פנים משקפיים וכו', בפרט, האדם המצולם אינו משתף פעולה עם הצלם ולאו דווקא מודע לכך שהוא מצולם. התמונות שצריך לזהות הן תמיד תמונות שאין עבורם ידע מוקדם. כלומר, גם אם תמונה של אדם כלשהו התקבלה יותר מפעם אחת כקלט, היא עדיין זרה לגמרי עבור המערכת משום שהיא אינה משתפרת מזיהוי לזיהוי.

תכונות אלה מאפשרות לבעיה להיות שימושית עבור מערכות הדורשות זיהוי ממוחשב שכן תנאי הצילום במערכות אמיתיות אינם קבועים, התמונות לרוב אינן מוכרות, והזיהוי צריך להתמבצע בזמן אמת. לכן, בשל העניין הגובר בפיתוח מערכות כאלה, העיסוק בבעיה הספציפית של זיווג תמונות נטולות אילוצים של פנים הוא בעל חשיבות גבוהה.

6.1 מאגר הנתונים LFW

השיטות שנתאר נבדקו על מאגר הנתונים LFW – Labeled Faces in the Wild [1], המכיל תמונות שנאספו מהאינטרנט ועל כן מכילות גיוון בתכונות כמו תאורה, זוויות פנים, הבעות וכו' הדומה באופן יחסי לגיוון הקיים במציאות. מאגר זה הוא אחד מן מאגרי הנתונים המקובלים היום לבדיקת ביצועים של זיווג פנים, ומשמש כמאגר הפנים עבור תחרות להשגת תוצאות גבוהות של זיווג פנים. LFW מכיל 13,233 תמונות פנים של 5749 אנשים שונים. רוב התמונות הן תמונות צבע אך ישנן גם כמה תמונות בגווני אפור. חלק מהתמונות מכילות כמה פנים אך עבור כל תמונה כזאת, מתייחסים רק אל הפנים שהפיקסל המרכזי של התמונה מוכל בהן. עבור כל תמונה נתון השם של האדם המופיע בה ומספר המציין את מספר התמונה שלו מתוך רשימת התמונות שלו (למשל George_W_Bush 1 היא התמונה הראשונה של ג'ורג' וו. בוש, George_W_Bush 2 היא התמונה השניה שלו וכו').

התמונות במאגר הנתונים התקבלו מתוך קובץ של תמונות מתוך חדשות של אתר האינטרנט Yahoo ([47]) שנאספו בשנים 2002-2003. על מאגר זה של בערך חצי מיליון תמונות, הורץ אלגוריתם לאיתור פנים של ספריות הקוד הפתוח OpenCV [48] שמתבסס על האלגוריתם של ויולה וג'ונס (Viola and Jones [2]). לאחר מכן הופעל תהליך ידני על-מנת לוודא שכל התמונות אכן מכילות פנים. כמו-כן תהליכים ידניים הורידו תמונות כפולות וצירפו את התגים עם השמות לתמונות. תהליך זה של יצירת מאגר הנתונים מבטיח מגוון רחב של תנאי צילום, הבעות, זוויות פנים וכו'.

האלגוריתם לאיתור הפנים של ויולה וג'ונס מחזיר חלון ריבועי מסביב לפנים שנמצאו. מגדילים חלון זה פי 2.2 בכל מימד והתמונה המתקבלת היא החלק של התמונה המופיע בחלון זה. אם

החלון חורג מגבולות התמונה מרפדים אותה בפיקסלים שחורים כנדרש. דוגמאות לתמונות המופיעות במאגר ניתן לראות באיור 4.

איור 4 – תמונות ממאגר הנתונים LFW

שלושת הזוגות החיוביים הראשונים במאגר (מצד שמאל) ושלושת הזוגות השליליים הראשונים במאגר (מצד ימין).

המרת תמונות הפנים לצורה מיוצבת (aligned), או לצורה קנונית (canonical pose) קבועה בה איברי הפנים ממוקמים בקואורדינטות קבועות, משפרת את תהליך הזיהוי משום ששימוש בתמונות מיוצבות מצמצם את המגוון האפשרי של התמונות איתם הוא צריך להתמודד ועוזר להיפטר מפרטים לא חשובים בתמונה. למעשה ייצוב תמונות הוא תת-תחום חשוב של זיהוי פנים ([49]). הסעיפים הבאים מתארים שתי גרסאות מיוצבות של LFW.

6.2 הגרסא הממוקדת (funneled version)

אחת הבעיות בייצוב של תמונות היא שאלגוריתמים של ייצוב בדרך-כלל מתאימים רק לסוג מסוים של אובייקטים. מיקוד היא שיטת ייצוב כללית [50] שיכולה לפעול על כל סוגי האובייקטים. מכאן המונח מיקוד (funneling) שהוא שונה מהמונח ייצוב (alignment) המקובל יותר בייצוב פנים. תהליך המיקוד מבוסס בעיקר על תהליך הנקרא הקפאה

(congealing). תהליך זה משתמש בלמידה בלתי מונחית (unsupervised learning). כלומר, לא נתונה קבוצת למידה ולא נדרש כל תהליך ידני מקדים. לצורך תיאור השיטה, נגדיר תחילה כמה מושגים:

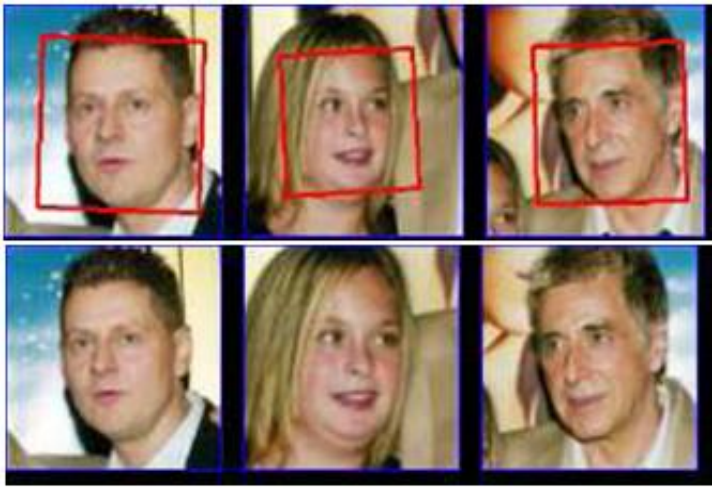
מחסנית פיקסלים היא קבוצת ערכי כל הפיסקלים המופיעים בקבוצה של תמונות במיקום מסוים. שדה ההסתברות הוא פונקציית ההתפלגות מעל קבוצה זו עבור כל מיקום אפשרי של פיקסל בתמונה. ניתן להסתכל על כך באופן הבא: אם יש j פיקסלים בתמונה, אזי נסמן ב-X_i, $1 \leq i \leq j$, את המשתנה המקרי המתאים למיקום i. ערכיו האפשריים של כל משתנה מקרי הם מחסנית הפיקסלים שלו. שדה ההסתברות הוא פונקציה המתאימה לכל X_i את ההתפלגות שלו מעל מחסנית הפיקסלים.

האנטרופיה האמפירית (empirical distribution) של פיקסל כלשהו היא האנטרופיה (ראה 14.2.4), כלומר מידת החוסר וודאות, של ההתפלגות האמפירית של פיקסל כלשהו מעל מחסנית הפיקסלים שלו.

בתהליך ההקפאה מפעילים באופן איטרטיבי את תהליך הבא: מחשבים את סכום האנטרופיות האמפיריות של כל הפיקסלים. עבור כל תמונה בוחרים טרנספורמציה מקבוצה נתונה של טרנספורמציות (בדרך כלל טרנספורמציות אפיניות) כך שערכו של סכום האנטרופיות יצומצם לאחר הפעלת הטרנספורמציות שנבחרו על התמונות. מאחר ואלה הן טרנספורמציות אפיניות התמונות לא ישתנו באופן מהותי. פרטים נוספים על השיטה ניתן למצוא ב-[50],[51].

לאחר שהתקבלה קבוצת תמונות חדשה, ובהינתן תמונה חדשה, ניתן לצרפה לקבוצת התמונות ושוב להפעיל את התהליך על כל התמונות. ניתן לייעל שיטה זו ע"י שמירה של כל ההתפלגויות האמפיריות המתקבלות בכל איטרציה. אזי בהינתן תמונה חדשה, אפשר להפעיל עליה את התהליך האיטרטיבי בהסתמך על ההתפלגויות שנשמרו.

שיטה זו מתגלה כבעייתית כשמנסים אותה בתנאים מציאותיים, למשל על תמונות הפנים של מאגר הנתונים LFW. זאת משום שרקעים שונים של התמונות, או תאורות שונות, ייצרו ערכי אנטרופיה גבוהים גם כאשר התמונות מיוצבות.
פתרון אפשרי לכך הוא במקום להשתמש בערכי הצבע של הפיקסלים עבור מרחב ההסתברות, להשתמש בערכים של מתארי SIFT [44] בגודל 8X8 המופעלים עבור כל פיקסל. מאחר שמתאר SIFT כנ"ל הוא וקטור בן 32 מימדים, מרחב ההסתברות הנוצר הוא גדול ומאט את החישובים ולכן מצמצמים אותו ע"י חלוקה (clustering) באמצעות שיטה הנקראת k-means (ראה 7.1.1). דוגמאות לתמונות לפני ואחרי הקפאה ניתן לראות באיור 5.



איור 5 – תהליך ההקפאה

תמונות אחרי הרצה של אלגוריתם לאיתור פנים (שורה עליונה) ותמונות לאחר תהליך ההקפאה (שורה תחתונה). כמה שינויים שניתן לזהות בעין הם: הפנים של האשה בתמונה האמצעית הוגדלו, הרקע של הפנים בתמונה הימנית השתנה והפנים הוזזו שמאלה, כמו כן גם הפנים בתמונה השמאלית.

LFW-a 6.3

LFW-a היא קבוצת התמונות הנוצרת ע״י שיטת ייצוב שונה משיטת המיקוד שהוסברה בסעיף הקודם. שיטת ייצוב זו מבוססת על שני שלבים: בשלב הראשון, משתמשים בתהליך למידה מבוקר (supervised learning) שמטרתו לאתר 9 נקודות השוואה (fiducial points) בפנים אשר מהוות נקודות זיהוי חשובות [46] (ראה איור 6). נקודות אלה מאותרות בדומה לשיטת איתור הפנים של ויולה וג׳ונס, כאשר במקום פנים מאותרים חלקי פנים הדרושים לצורך סימון הנקודות (למשל קצה העין הימנית, אמצע האף וכדומה). בשלב השני, מעבירים את התמונות טרנספורמציות דמיון הממפות את נקודות ההשוואה שלהן לקואורדינטות קבועות נתונות. פרטי האלגוריתם אינם ידועים משום שהוא מסחרי. נמצא שיש שיפור משמעותי ביכולת זיהוי הפנים כאשר משתמשים ב-LFW-a לעומת שימוש בשיטת המיקוד [46] וניתן לראות באיור 7 תמונת פנים המדגימה זאת.

איור 6 – נקודות השוואה

תמונת פנים לאחר מציאת נקודות ההשוואה ע״י האלגוריתם של ויולה וגו׳נס.

איור 7 – LFW-a לעומת שיטת המיקוד

משמאל, תמונת פנים מיוצבת לפי שיטת LFW-a, מימין תמונת פנים מיוצבת בשיטת המיקוד. ניתן לראות שבתמונה השמאלית איברי הפים קרובים יותר לנקודות לפיהן מבצעים את הייצוב מאשר בתמונה הימנית.

6.4 תהליכי הלמידה והבחינה ועקרון המניעה ההדדית

עקרון חשוב של LFW הוא מניעה הדדית בין קבוצת הלמידה לקבוצת הבחינה. כלומר, כל תמונה המשמשת כקלט בזמן הבחינה אינה מופיעה בקבוצת הלמידה. כך בתהליך הלמידה אין אפשרות ללמוד מודל פנים שיזוהו בתהליך הבחינהוכל תמונות הפנים בתהליך הבחינה אינן מוכרות ומכילות פנים של אנשים שאינם מוכרים. מכאן שמטרת הלמידה היא ללמוד להבחין בין פנים של אנשים שונים ופחות ללמוד איך לאפיין אדם לפי מאגר תמונות שלו. בפועל, עקרון זה לא ממומש לחלוטין אך נשמר באופן סביר תחת שני סוגים של הגדרות סביבה (views):

1. סביבה ראשונה (view 1): משמשת לפיתוח האלגוריתם. בסביבה זו התמונות מתחלקות לשתי קבוצות זרות. קבוצת הלמידה מכילה 1100 זוגות של תמונות פנים כשכל זוג הוא זוג תואם (מכיל פנים של אותו אדם) ו-1100 זוגות של תמונות פנים שאינם תואמים (מכילים פנים של אנשים שונים). קבוצת ההרצה מכילה 500 זוגות של תמונות פנים תואמים ו-500 זוגות של תמונות פנים לא תואמים. בסביבה זו נשמר עקרון המניעה ההדדית משום שקבוצות הלמידה והבחינה זרות. סביבה זו היא היחידה בה ניתן לשנות את האלגוריתם.
2. סביבה שניה (view 2): משמשת לדיווח על תוצאות האלגוריתם. השאיפה היא שהאלגוריתם ירוץ רק פעם אחת בסביבה זו על-מנת למדוד את ביצועיו. כלומר, בשלב הראשון משתמשים בסביבה הראשונה בכדי לעצב את האלגוריתם ולפתחו. בשלב הבא, משתמשים בסביבה השניה בשביל למדוד את הישגיו. בסביבה השניה מחולק מאגר התמונות ל-10 קבוצות זרות. אופן ההרצה הוא כלדהלן: מריצים את האלגוריתם 10 פעמים כשבריצה ה-i קבוצת הבחינה היא הקבוצה ה-i וקבוצת הלמידה היא איחוד כל שאר הקבוצות. למשל, אם נמספר את הקבוצות מ-1 עד 10, בריצה הראשונה קבוצת הלמידה היא (2,3,4,5,6,7,8,9,10) והבחינה מתבצעת על קבוצה 1. בריצה הרביעית קבוצת הלמידה היא (1,2,3,5,6,7,8,9,10) והבחינה מתבצאת על קבוצה 4. התוצאות המדווחות

הן הדיוק הממוצע שהושג ב-10 ריצות אלה וסטיית התקן של הממוצע. ליתר דיוק, אם p_i הוא אחוז הדיוק בריצה i, אז הערכים המוחזרים הם:

- הדיוק הממוצע:

$$\hat{\mu} = \frac{\sum_{i=1}^{10} p_i}{10} \quad (1)$$

שגיאת התקן של הממוצע:

$$S_E = \frac{\hat{\sigma}}{\sqrt{10}} \quad (2)$$

כאשר $\hat{\sigma}$ היא סטיית התקן המדגמית (ראה 14.2.3):

$$\sqrt{\frac{\sum_{i=1}^{10}(p_i - \hat{\mu})^2}{9}} \quad (3)$$

למרות שעקרון המניעה ההדדית אינו נשמר כאן באופן מלא (למשל תמונות שהופיעו בקבוצת הלמידה בסביבה הראשונה מופיעות גם בקבוצות הבחינה בסביבה השניה), ההנחה היא שתמונות בקבוצות הבחינה ייחשבו כמו תמונות שאינן מוכרות משום שאלגוריתם שמתאים את עצמו לקבוצות הבחינה בשלב הלמידה הוא אלגוריתם שלא יעבוד טוב בסביבה הראשונה בה כן נשמר עקרון המניעה ההדדית. כלומר, מסתמכים על העובדה שאלגוריתם שעובר את השלב הראשון בהצלחה הוא אלגוריתם כזה שמניח שתמונות בשלב הלמידה לא יופיעו בשלב הבחינה.

6.5 הגישה המוגבלת והגישה שאינה מוגבלת בקבוצת הלמידה

קבוצת למידה של LFW מכילה זוגות של תמונות פנים כאשר לכל זוג יש תג הקובע האם הוא זוג חיובי (פנים של אותו אדם) או אם זהו זוג שלילי (פנים של אנשים שונים). למידה במאגר זה יכולה להתבצע בשתי דרכים אפשריות:

1. באופן בלתי מוגבל (unrestricted training) – כל זוג חיובי מקבל כתיוג את שמו של האדם שפניו מופיע בתמונות אלה. מכאן שאם למשל שני הזוגות (1,2) ו-(3,4) מתויגים כפנים של George_W_Bush, אז המשתמש רשאי לייצר את הזוגות הנוספים (1,3), (1,4), (2,3) ו-(2,4) ולהוסיפם אל קבוצת הלמידה.
2. באופן מוגבל (image-restricted training) - זוגות של תמונות מסומנים בשני סימונים אפשריים – זוג חיובי (פנים של אותו אדם) או זוג שלילי (לא אותו אדם). מתייחסים אל הזוגות כאילו שמותיהם של האנשים המופיעים בהם אינם ידועים. במקרה כזה אין זוגות חיוביים נוספים הנובעים מקבוצת הזוגות הקיימת.

כל שיטה המציגה את ביצועיה על מאגר הנתונים LFW צריכה לציין האם היא השתמשה במאגר באופן מוגבל או בלתי מוגבל.

7 אלגוריתמי חלוקה וסיווג שימושיים

בסעיף זה נתאר כמה אלגוריתמי למידה שימושיים המוזכרים בשיטות הזיווג בהם נעסוק בסעיפים הבאים. נדון בשני סוגים של בעיות: בעיות סיווג (classification) ובעיות חלוקה (clustering).

7.1 חלוקה (clustering)

בבעיות חלוקה המטרה היא לחלק נתונים לקבוצות הנקראות מחלקות (clusters) לפי מידת הדימיון שלהם אחד לשני.

7.1.1 חלוקה ל-k ממוצעים (k-means clustering)

אלגוריתם למידה לא מונחה זה מנסה לחלק n נתונים $(x_1, \dots, x_n)$ ל-k מחלקות $(k < n)$ באופן שיהיה כמה שפחות פיזור של הנתונים בתוך המחלקות. כלומר השאיפה $S = \{S_1, \dots, S_k\}$ היא להביא למינימום את הסכום:

$$\sum_{i=1}^{k} \sum_{k_j \in S_j} \|k_j - \mu_i\|^2 \qquad (4)$$

כאשר μ_i הוא הממוצע של הנקודות במחלקה S_i.

האלגוריתם הסטנדרטי בו משתמשים לחישוב החלוקה מחליט (בדר"כ באופן אקראי) על k ממוצעים התחלתיים, ומחלק את כל הנתונים כך שכל נתון יהיה במחלקה של הממוצע שהכי קרוב אליו. לאחר מכן הוא ממשיך באופן איטרטיבי לחשב בכל פעם את הממוצעים החדשים של המחלקות, ושוב לחלק את הנתונים למחלקה עם הממוצע הקרוב ביותר. כאשר מתרחשת התכנסות (כאשר לאחר חישוב הממוצעים מחדש אין צורך לשנות את החלוקה כי כל נתון כבר נמצא במחלקה שלה יש את הממוצע הקרוב אליו ביותר) המחלקות שהתקבלו מהוות את החלוקה הסופית.

7.2 אלגוריתמי סיווג (classification) שימושיים

בבעיות סיווג המטרה היא לתייג נתון חדש כשייך לקבוצה מסוימת (הנתון מקבל תג שהוא שם המאפיין אותו), לפי התכונות שלו ולפי קבוצת למידה שתויגה כבר לקבוצות האפשריות הקיימות. כלומר, בניגוד לאלגוריתמי חלוקה מהסעיף הקודם, אלגוריתמי סיווג משתמשים בלמידה מונחית. האלגוריתם שנתאר בסעיף זה הם אלגוריתם ה-k שכנים הקרובים ביותר (k-nearest neighbor – kNN), Adaboost ו-SVM.

מסווג (classifier) הוא פונקציה שהקלט שלה הוא תמונה, והפלט שלה היא תוצאה של סיווג, כלומר התג המתאים לתמונה זו. מסווג בינארי הוא מסווג שהפלט שלו הוא 0 או 1.

7.3 k השכנים הקרובים ביותר (k-nearest neighbor – kNN)

אלגוריתם kNN הוא אלגוריתם סיווג בסיסי ואינטואיטיבי הקובע סיווג לפי החלטת הרוב של השכנים: בהינתן קבוצת למידה של n דוגמאות מסווגות וקלט חדש שאינו מוכר, האלגוריתם ימצא (למשל באמצעות מרחק אוקלידי) את k הדוגמאות הקרובות ביותר אליו מבין כל n הדוגמאות. הסיווג השכיח ביותר מבין k הסיווגים של שכנים אלה ייבחר כסיווג המתאים עבור הקלט.

7.4 Adaboost

Adaboost הוא אלגוריתם סיווג בינארי שפותח ע"י Freund and Schapire [52] ב-1996. האלגוריתם בונה מסווג מתוך קבוצה גדולה של מאפיינים, המתבסס על קבוצה קטנה של מאפיינים שנבחרים מהקבוצה ההתחלתית. התהליך הוא איטרטיבי כאשר בכל פעם נבחר עוד מאפיין שעליו מתבסס המסווג, עד שנבחר המספר הרצוי של המאפיינים עליהם בוחרים להתבסס. נמשיך בתיאור יותר מפורט של תהליך זה, ונתחיל בכמה הגדרות:

קבוצת הלמידה של האלגוריתם היא קבוצה של דוגמאות מסווגות $(x_1, y_1), \dots, (x_n, y_n)$, כאשר x_i מייצג תמונה וy_i- הוא 0 או 1. השגיאה של מסווג h מחושבת עבור קבוצת למידה נתונה ועבור וקטור משקלים נתון w המתאים משקל w_i לכל דוגמא. והיא הסכום של השגיאות עבור כל דוגמא ממושקל במשקלים אלה, כלומר:

$$\sum_i w_i |h(x_i) - y_i| \qquad (5)$$

תהליך לומד חלש (weak learner) הוא מסווג אשר מצליח לסווג תמונה באופן יותר טוב מאשר החלטה אקראית. ליתר דיוק, קיימים $0 < \delta < 1, \varepsilon < \frac{1}{2}$, כך שבהסתברות של לפחות $1 - \delta$, ובהינתן תמונה כלשהי, ההסתברות של המסווג לטעות בסיווגה אינה גבוהה מ-ε (כלומר אחוז השגיאה קטן בערך כלשהו מ-$\frac{1}{2}$).

תהליך לומד חזק (strong learner) הוא תהליך המסוגל לייצר עבור **כל** $0 < \delta < 1, \varepsilon < \frac{1}{2}$, מסווג אשר בהינתן תמונה כלשהי, ההסתברות שלו לטעות בסיווגה אינה גבוהה מ-ε.

האלגוריתם Adaboost מייצר מסווג חזק המורכב מצירוף לינארי של מספר מסווגים חלשים, ניתן לקבוע באופן ידני פרמטרים שונים השולטים על מידת הדיוק מול מהירות הריצה וגם על כמה מסווגים חלשים המסווג הסופי מסתמך. בניית המסווג החזק נעשית באופן הבא: הקלט הוא קבוצת למידה ווקטור משקלים אשר מאותחל למשקלים כלשהם המסתמכים על אחוז הדוגמאות החיוביות והשליליות בקבוצת הלמידה. משקלים אלה נמצאו ע"פ ניסויים כמשקלים היעילים ביותר. בשלב הראשון, האלגוריתם משתמש במספר איטרציות מספיק גדול בשביל להשיג את

מטרות הדיוק שלו מהמסווג הסופי. בכל איטרציה, מייצרים תהליך לומד חלש עבור כל מאפיין בקבוצת המאפיינים ומחשבים את השגיאה של כל מסווג על קבוצת הלמידה לפי וקטור המשקלים. לאחר מכן בוחרים את המסווג החלש בעל השגיאה הכי נמוכה. את המשקלים מעדכנים כך שהדוגמאות עליהן המסווג הנ״ל טעה מקבלות משקל גבוה יותר. באופן איטרטיבי ממשיכים לבחור מסווגים חלשים כמידת הצורך ומרכיבים את כולם בנוסחא המהווה את המסווג החזק. נוסחא זו בנויה כך שהמסווג שנוצר אכן יהיה מסווג חזק אשר אחוז השגיאה שלו מתקרב לאפס באופן אקספוננציאלי כתלות במספר האיטרציות. כמו-כן מסווגים חלשים בעלי אחוז שגיאה נמוך יותר מקבלים מקדמים גבוהים יותר במשוואה ולכן ישפיעו יותר על המסווג החזק. נשים לב גם לכך שבעצם נעשתה בחירה של מספר קטן של מאפיינים מתוך קבוצה גדולה של מאפיינים אשר מהווים את המאפיינים הכי מתאימים לסיווג. כלומר תוצאת Adaboost היא גם בחירה של קבוצה קטנה של מאפיינים אשר אפקטיביים במיוחד עבור סיווג רצוי כלשהו, וגם מסווג חזק המבוסס עליה. האלגוריתם מפורט באיור 8.

7.4.1 בחירה מראש של מאפיינים (forward feature selection)

אלגוריתם ה-Adaboost שתואר מקודם מייצר תהליכים לומדים חלשים עבור כל מאפיין בכל איטרציה. כמו-כן הוא מחשב את כל המשקלים החדשים עבור כל הדוגמאות בכל איטרציה. תהליכים אלה עלולים להיות מאוד איטיים, במיוחד כאשר יש מספר רב של מאפיינים ודוגמאות. אחד מהפתרונות האפשריים לבעיה זו הוא בחירה מראש של מאפיינים (forward feature selection, [53]) אשר בה כל המסווגים מיוצרים פעם אחת בהתחלה, תוצאותיהם על כל דוגמא מוכנסות לטבלה גדולה, מיושמת באופן כזה שהחיפוש בה מהיר. בכל איטרציה עוברים על כל המסווגים ובוחרים את המסווג אשר ישפר במידה הכי גדולה את המסווג הסופי שיורכב מהצבעת הרוב של המסווגים שנבחרים. פסאודו קוד לאלגוריתם מפורט באיור 9. אלגוריתם זה נמצא מהיר בערך פי 100 מהאלגוריתם Adaboost.

7.5 מכונת וקטורים תומכים – Support Vector Machine - SVM

מכונת וקטורים תומכים היא שיטה שהוצגה ע״י ולדימיר ופניק ב-1963 [54] ומאז מהווה כלי מרכזי וחשוב לפתרון בעיות סיווג. מכונת וקטורים תומכים היא מסווג היוצר הפרדה בין הוקטורים המייצגים את התמונות במרחב השייכות לקבוצת הלמידה באמצעות מישור-על (hyperplane). המישור-על שנוצר מפריד לינארית בין הדוגמאות, שני מישורי העל המוגדרים כתוצאה מהפרדה זו מתלכדים עם הדוגמאות (איור 10), אחד עם הדוגמאות החיוביות ואחד עם השליליות. מטרת ה-SVM היא להגדיל את המרחק בין המפריד הלינארי ובין כל אחד מהמישורים המתלכדים עם הדוגמאות - מרחק המכונה שול (margin) - ולמעשה למצוא את המפריד בעל השוליים הרחבים ביותר. ניתן להגדיר את מישור ההפרדה לפי דוגמאות האימון המתלכדות עם מישורי השוליים הנקראות וקטורים תומכים (support vectors).

- קלט: קבוצת למידה של דוגמאות מסווגות $(x_1, y_1), \dots, (x_n, y_n)$, כאשר x_i מייצג תמונה ו-y_i הוא 0 או 1.
- איתחול המשקלים – עבור כל דוגמא בקבוצת הלמידה:

$$w_{1,i} = \begin{cases} \frac{1}{2m} & y_i = 0 \\ \frac{1}{2l} & y_i = 1 \end{cases}$$

כאשר m הוא מספר הדוגמאות השליליות ו-l הוא מספר הדוגמאות החיוביות.

- עבור $t = 1, \dots, T$ (T הוא מספר המסווגים החלשים עליהם נרצה להתבסס):
  1. נרמול המשקלים כך שהסכום שלהם יהיה שווה ל-1:

$$w_{t,i} = \frac{w_{t,i}}{\sum_{j=1}^{n} w_{t,j}}$$

  2. עבור כל מאפיין j, צור מסווג חלש C_j, המתבסס עליו. חישוב השגיאה שלו ביחס לוקטור המשקלים w:

$$\epsilon_j = \sum_i w_i |h(x_i) - y_i|$$

  3. בחירת המסווג החלש, C_t, בעל השגיאה הנמוכה היחידה.
  4. עדכון המשקלים:

$$w_{t+1,i} = w_{t,i} \beta_t^{1-e_i}$$

כאשר $e_i = 0$ אם דוגמא x_i סווגה נכון ו-$e_i = 1$ אחרת ו-$\beta_t = \frac{\epsilon_t}{1-\epsilon_t}$.

- המסווג החזק הסופי הוא:

$$H(x) = \begin{cases} 1 & \sum_{t=1}^{T} \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0 & \text{אחרת} \end{cases}$$

איור 8 - Adaboost

אלגוריתם ה-Adaboost עבור יצירת מסווג חזק מתוך קבוצה גדולה של מאפיינים.

ובאופן פורמלי: נתונה קבוצת הלמידה המכילה את הדוגמאות x_i, $1 \leq i \leq n$, כל דוגמא מסווגת באמצעות תג $c_i = 1$ או $c_i = -1$. על-מישור יכול להיות מוגדר כקבוצת הנקודות x המקיימות $w \cdot x - b = 0$, כאשר רוחב השול הוא $\frac{1}{\|w\|}$. ההיסט של העל-מישור מראשית הצירים בכיוון w הוא $\frac{b}{\|w\|}$. המטרה היא אם-כן לבחור את b ו-w כך שהעל-מישור באמת יפריד בין הדוגמאות, כלומר שלכל הדוגמאות השליליות יתקיים $w \cdot x - b \leq -1$ ושלכל הדוגמאות החיוביות יתקיים $w \cdot x - b \geq 1$. כמו-כן השאיפה היא ליכולת הכללה כמה שיותר גדולה, כלומר, שהשול יהיה כמה שיותר רחב. זאת משיגים ע"י מציאת ערך $\|w\|$ מינימלי.

- קלט: קבוצת למידה של דוגמאות מסווגות $(x_1, y_1), \ldots, (x_n, y_n)$, כאשר x_i מייצג תמונה ו-y_i הוא 0 או 1. d הוא אחוז הזיהוי המינימלי המבוקש, ו-f הוא אחוז השגיאה מסוג I המקסימלי (false positive)
- עבור כל מאפיין j, ייצר מסווג חלש C_j אשר אחוז השגיאה מסוג I שלו לא גדול מ-f.
- אתחל את קבוצת המסווגים לקבוצה ריקה: $H \leftarrow \phi, t \leftarrow 0, d_0 \leftarrow 0, f_0 \leftarrow 0$.
- כל עוד $d_t < d$ או $f_t > f$:
  1. אם $d_t < d$, מצא את המאפיין k, אשר אם נוסיף אותו ל-H, נשיג אחוז דיוק גבוה יותר מ-d_t אשר נסמנו ב-d_{t+1}.
  2. אחרת, מצא את המאפיין k, אשר אם נוסיף אותו ל-H, נשיג אחוז שגיאה מסוג I נמוך יותר מ-f_t אשר נסמנו ב-f_{t+1}.
  3. $H \leftarrow H \cup \{h_k\}, t \leftarrow t + 1$
- המסווג החזק הסופי מתקבל מהחלטת הרוב של המסווגים החלשים שנמצאים ב-H. כלומר:

$$H(x) = \begin{cases} 1 & \sum_{h_j \in H} h_j(x) \geq \theta \\ 0 & \text{אחרת} \end{cases}$$

כאשר $\theta = \frac{T}{2}$. ניתן לבחור ערך יותר נמוך במקרה הצורך.

איור 9 – בחירת מאפיינים מראש

פסאודו קוד של תהליך בחירת מאפיינים מראש כתחליף מהיר יותר לאלגוריתם adavoost.

לסיכום, בכדי למצוא על-מישור שיפריד לינארית בין הדוגמאות בקבוצת הלמידה יש לפתור את בעיית המינימיזציה:

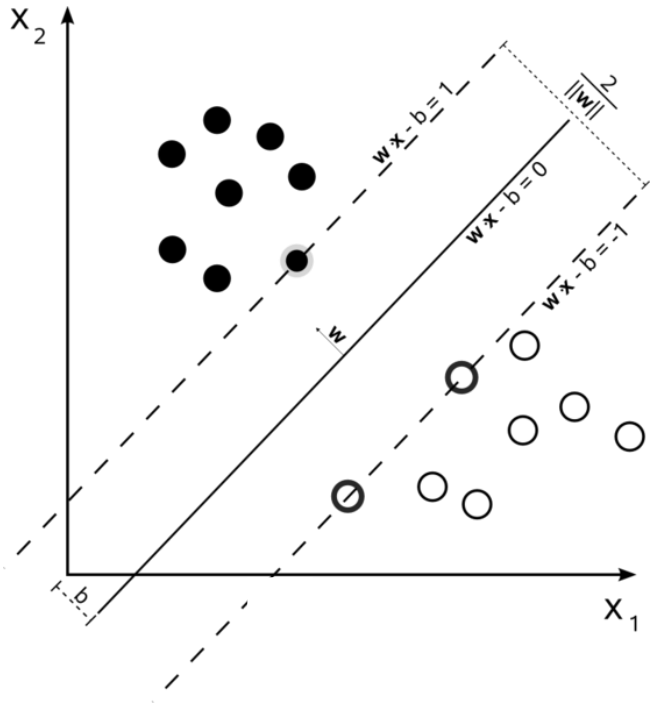
$$Minimize \ \|w\|$$

$$s.t \ (for \ any \ i = 1, \ldots, n)$$

$$w \cdot x - b \leq -1$$

$$w \cdot x - b \geq 1$$

$\|w\|$ היא נורמה ועל-כן מכילה את פעולת השורש הנחשבת לפעולה יקרה. לכן, רוצים לחפש את $\|w\|^2$ במקום $\|w\|$. כמו-כן מאחדים את שתי האי-שיוויונות לאי-שוויון אחד ע"י הגדרת משתנה חדש c_i שמייצג סימן. מקבלים את הבעיה השקולה:



איור 10 – הפרדה ע"י על-מישור

על-מישור מפריד בין דוגמאות חיוביות (לבנות) ושליליות (שחורות) בשני מימדים בעל השוליים הרחבים ביותר. הנקודות הנמצאות על קצה השוליים הם הוקטורים התומכים. המישור שמתחת לקו ההפרדה מתלכד עם כל הדוגמאות החיוביות והמישור שמעל מתלכד עם כל הדוגמאות השליליות.

$$Minimize \ \|w\|^2$$

$$s.t \ (for \ any \ i = 1, \dots, n)$$

$$c_i(w \cdot x - b) - 1 \geq 0$$

ע"י שימוש בכופלי לגרנז' (Lagrange multipliers [55]), w יכול להיות מבוטא באמצעות הוקטורים התומכים x_i, וכופלי לגרנז' α_i באופן הבא: $w = \sum_{i=1}^{n} \alpha_i c_i x_i$.

באמצעות שיטה הנקראית תכנות ריבועי (quadric programming [56]) מוצאים את w ו-b וכך מקבלים מכונת מסווגת – בהינתן דוגמא x, ניתן להציבה בנוסחא $\sum_{i=1}^{n} \alpha_i c_i x_i \cdot x - b$. עם התוצאה קטנה או שווה ל-(1-), זוהי דוגמא שלילית. אם התוצאה גדולה או שווה ל-1, זוהי דוגמא חיובית.

על מנת להתמודד עם הסתירה המתעוררת לפעמים בין הניסיון לשמור על שוליים רחבים מצד אחד (ועל ידי כך להביא להפרדה כללית יותר) ובין הנסיון לסווג כמה שיותר דוגמאות בצורה נכונה מצד שני, מגדירים משתנים הנקראים משתני סרק (slack variables) המתארים את מרחקי השגיאות של נקודות אותן לא מצליחים לסווג, ומגדירים קבוע ענישה (penalty) המסומן לרוב באות C. ערך גבוה פירושו העדפה של סיווג נכון וערך נמוך - העדפה של שוליים רחבים.

7.5.1 הפרדה לא לינארית

במקרים בהם הנתונים מפוזרים במרחב באופן כזה שהם אינם ניתנים להפרדה בצורה לינארית (תופעה זו היא שכיחה ברוב בעיות הסיווג המעניינות), משתמשים במיפוי של הנתונים למרחב רב מימדים הנקרא מרחב המאפיינים (feature space) אשר בו לרוב קל יותר למצוא מפריד לינארי מתאים. פונקציית המיפוי מסומנת בדר״כ ב-φ ולכן מכונת ה-SVM עם מיפוי כנ״ל נראית כך:

$$\left(\sum_{i=1}^{n} \alpha_i c_i \varphi\left(x_i\right)\right) \cdot \varphi\left(x\right) - b = \sum_{i=1}^{n} \alpha_i c_i \varphi\left(x_i\right) \cdot \varphi\left(x\right) - b \quad (6)$$

השווין הנ״ל מתאפשר מכיוון שהוקטורים התומכים ידועים ומכאן שהביטוי שבסוגריים הוא לא משתנה. המכפלה הפנימית $\varphi\left(x_i\right) \cdot \varphi\left(x\right)$ יכולה להיות מוחלפת בפונקציה הנראית פונקציית גרעין (kernel function), תהליך אשר נקרא תעלול גרעין (kernel trick).

פונקציית גרעין היא פונקציה המקיימת עבור כל זוג נקודות במרחב הקלט של הבעיה:

$$k\left(x_i, x_j\right) = \varphi\left(x_i\right) \cdot \varphi\left(x_j\right) \quad (7)$$

קיימות כמה פונקציות גרעין שימושיות כמו:

- פונקציות גרעין פולינומיאליות: $k\left(x_i, x_j\right) = \left(x_i \cdot x_j\right)^d$,$k\left(x_i, x_j\right) = \left(x_i \cdot x_j + c\right)^d$
- גרעין RBF (Radial Base Function): $k\left(x_i, x_j\right) = exp\left(-\gamma \left\| x_i - x_j \right\|^2\right)$
- RBF גאוסיאני: $k\left(x_i, x_j\right) = exp\left(-\frac{\left\| x_i - x_j \right\|^2}{2\sigma^2}\right)$

אם-כן, כאשר מחפשים SVM מתאים לבעיה פרמטרים חשובים שקובעים את אופי מכונת ה-SVM הם קבוע הענישה C ופונקציית הגרעין k.

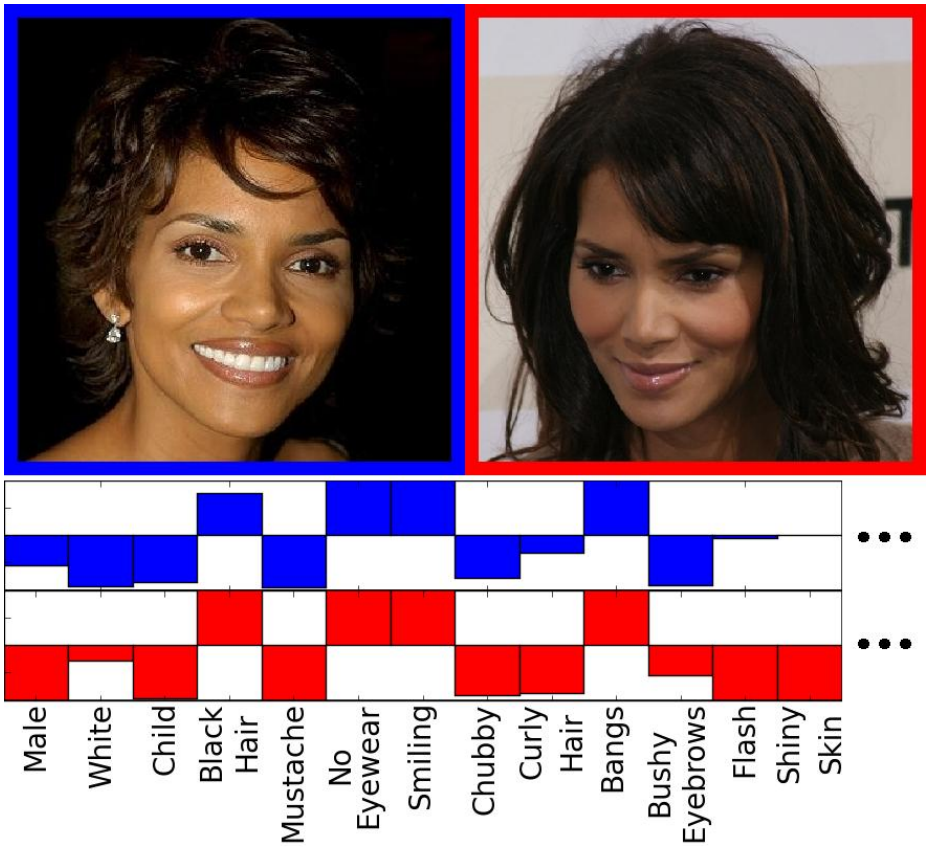
8 מסווגים של תיאורים ודימויים (Attribute and Simile Classifiers)

בפרק זה נדון בשתי שיטות שפותחו באוניברסיטת קולומביה ע״י קומר, ברג ועמיתיהם [57]. שיטות אלה לומדות ממאגר הנתונים LFW בגישה המוגבלת (ראה 6.5) וכל אחת מהן משיגה שיפורים גבוהים באחוזי ההצלחה של זיווג תמונות פנים. כאשר הן פועלות בשילוב יחדיו, הן משיגות אף שיפור גבוה יותר. המייחד את שיטות האלה הוא שימוש בתכונות (traits) ויזואליות שהן אינטואיטיביות לבני אדם כגון: זכר, שיער שחור, שפם, דימיון לאדם מתמונה אחרת וכדומה. משתמשים בתכונות ויזואליות מהסוג הזה בדרך כלל כשרוצים לחפש סוג מסוים של אנשים מתוך מאגר של תמונות (למשל אנשים מחייכים עם שפם) אך זאת כנראה הפעם הראשונה שמשתמשים בתכונות ויזואליות מהסוג הזה עבור זיהוי פנים. נפריד את התכונות לשני סוגים: תיאורים (attributes) – מאפיינים ויזואלים אינטואיטיביים כגון צבע שיער, צורת הפנים וכדומה, ודימויים (similes) – קביעה האם אזור פנים מסוים דומה או לא דומה לאדם כלשהו מקבוצת ייחוס נתונה.

השיטה הראשונה מתבססת על מסווגים של מאפייני תיאורים (attribute classifiers) שהם מסווגים בינאריים. השיטה מציגה 65 מסווגים כאשר ניתן בכל עת ליצור עוד מסווגים ולהוסיפם. המאפיינים עליהם מתבססים המסווגים הם ויזואליים ואינטואיטביים – מגדר, צבע השיער, גיל וכו׳. כל מסווג מתבסס על מאפיין כנ״ל וקובע תשובה חיובית או שלילית לשאלה האם הוא נוכח בתמונה. דוגמא לתוצאות של כמה מסווגים כאלה שהופעלו שתי תמונות של האלי ברי ניתן לראות באיור 11. נשים לב שבשתי התמונות תוצאות המסווגים דומות מלבד שימוש בפלאש בתמונה וגוון העור המבריק. כלומר רוב המסווגים סיווגו את התמונה באופן דומה, למרות שהיא מאוד שונה מבחינת הבעות, זווית, תאורה ואיכות תמונה.

ישנו שימוש בתיאורים שאינם מתארים פנים באופן ספציפי כמו פלאש משום ששיטה זו משמשת גם ככלי חיפוש של תמונות.

השיטה השניה מתבססת על מסווגים של דימויים (simile classifiers), גם הם מסווגים בינאריים הקובעים האם אזורים כלשהם של הפנים דומים לאזורים של הפנים של קבוצת אנשים ספציפית המשמשת כקבוצת הייחוס. שיטה זו מתבססת על הגישה שניתן לאפיין פנים של אדם כלשהו ע״י כך שנציין שהעיניים שלו נראות כמו העיניים של ריצ׳רד גיר, האף שלו נראה כמו האף של רוברט דה נירו וכדומה. בשיטה זו השתמשו בקבוצת ייחוס של 60 אנשים (שאינם מופיעים ב-LFW), כמובן שניתן להגדילה כמידת הצורך. דוגמא לכמה מסווגים של דימויים על תמונות של האריסון פורד ניתן לראות באיור 12. הסימון R_j מתייחס לאדם שמספרו j בקבוצת הייחוס, כך שלמשל העמודה הראשונה משמאל בדיאגרמה קובעת את הדימיון של העיניים בכל תמונה לעיניים של אדם שמספרו j בקבוצת הייחוס. נשים לב שרוב מסווגי הדימויים משיגים תוצאות דומות למרות השינויים בתאורה, בזוויות ובהבעות בין התמונות.

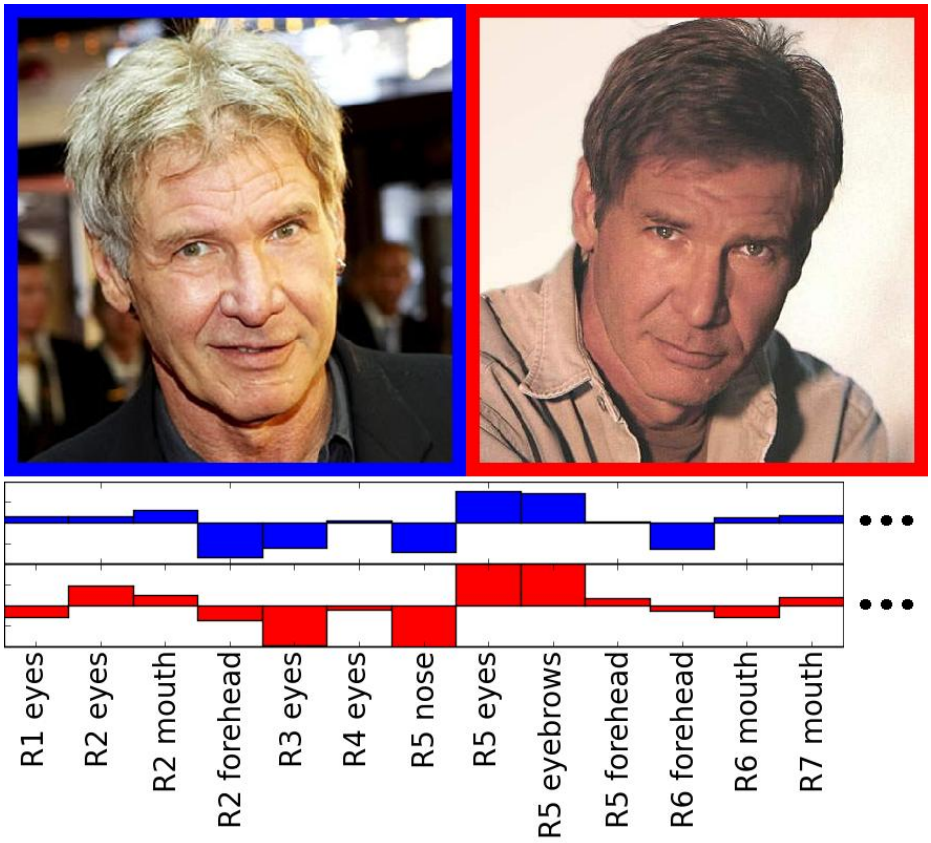


איור 11 – מסווגים של תיאורים

תוצאות של מסווגים של תיאורים על שתי תמונות של האלי ברי. העמודות האדומות מתייחסות לתמונה הממסוגרת באדום והעמודות הכחולות לתמונה הממסוגרת בכחול. ערך עמודה הוא חיובי וגבוה יותר ככל שיש סיכוי גבוה יותר שהמאפיין קיים בתמונה והוא שלילי ונמוך יותר ככל שיש סיכוי גבוה יותר שהוא לא קיים בתמונה. בשתי התמונות תוצאות המסווגים דומות מלבד הפלאש בתמונה וגוון העור המבריק. כלומר רוב המסווגים סיווגו את התמונה באופן דומה, למרות שהיא מאוד שונה מבחינת הבעות, זווית, תאורה ואיכות תמונה.

ההשוואה בין שתי תמונות פנים היא פשוטה קלה ואינה מצריכה חישובים יקרים – כל המסווגים, הן אלה המבוססים על תיאורים והן המבוססים על דימויים, מאורגנים בוקטורים שאינם ארוכים באופן יחסי (3000-65 מימדים). בשביל להשוות שתי תמונות של פנים, אפשר פשוט להתבסס על השוואות של הוקטורים המאפיינים אותן. זיווג באמצעות מסווגים של תיאורים ודימויים מאפשר חיפוש תמונות לפי תיאורים מסוימות (למשל ילד מחייך עם שיער כהה) או לפי דימיון (חפש את הידוענים שהכי דומים לי).

בפיתוח השיטות המתוארות בפרק זה, השתמשו במאגר נתונים נוסף ל-LFW הנקרא PubFig [58]. מאגר נתונים זה כולל 60,000 תמונות של 200 אנשים בלבד. לשם ההשוואה נזכיר כי ב-LFW ישנן 13,233 תמונות פנים של 5749 אנשים שונים. מספר התמונות השונות עבור כל אדם ב-PubFig מאפשר לימוד מתוך הרבה זויות אפשריות של פנים, הבעות ותנאי צילום שונים. שימוש משולב בשני מאגרי הנתונים LFW ו-PubFig מבטיח קבוצת לימוד רחבה עם מספר גדול של אנשים שונים.



איור 12 – מסווגים של דימויים

תוצאות של מסווגים של דימויים על שתי תמונות של האריסון פורד. הסימון $\boldsymbol{R_j}$ מתייחס לאדם שמספרו j בקבוצת הייחוס, כך שלמשל העמודה הראשונה משמאל בדיארגרמה קובעת את הדימיון של העיניים בכל תמונה לעיניים של אדם שמספרו j בקבוצת הייחוס. העמודות האדומות מתייחסות לתמונה הממסוגרת באדום והעמודות הכחולות לתמונה הממסוגרת בכחול. ערך עמודה הוא חיובי וגבוה יותר ככל שהדימיון קיים וגבוה יותר שלילי ונמוך יותר ככל שאזור הפנים שונה יותר מאזור הפנים של האדם אליו משווים. רוב מסווגי הדימויים משיגים תוצאות דומות למרות השינויים בתאורה, בזוויות ובהבעות בין התמונות.

תמונות הפנים במאגר Pubfig הן תמונות של ידוענים שנאספו באמצעות מנועי חיפוש של תמונות באינטרנט כמו google images ו-flicker.

בכל אחת משתי השיטות מייצרים בתהליך הלמידה קבוצת מסווגים בהם משתמשים מאוחר יותר בתהליך הבחינה. בנייתם של המסווגים ([59]) נעשית באותו אופן עבור דימויים ותיאורים ומתוארת בהמשך הפרק.

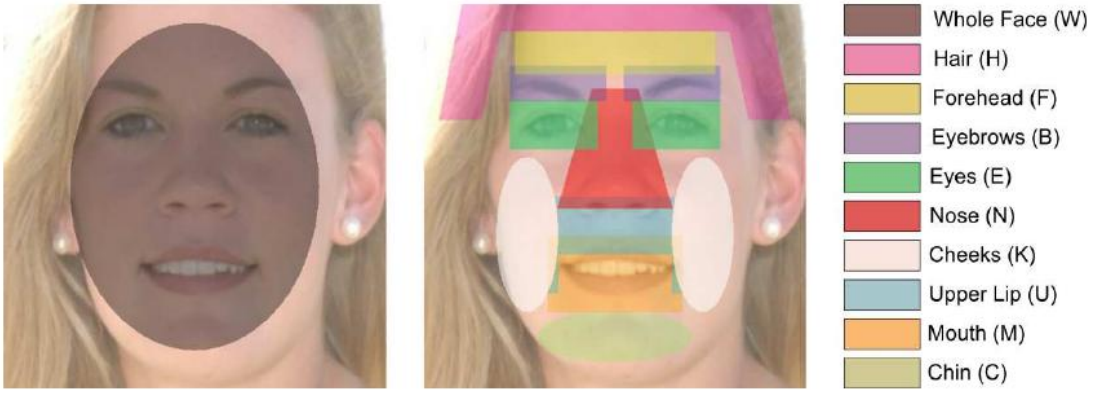
8.1 עיבוד ראשוני – ייצוב התמונות

בשלב הראשון, מפעילים מאתר פנים מסחרי שנקרא OKAO [60] על כל התמונות בקבוצת הלמידה. מאתר זה מחזיר את זוית הפנים ועוד 6 נקודות של הפנים שהן נקודות השוואה (ראה 5.1). נקודות אלה הן 4 הפינות של שתי העיניים, ו-2 הפינות של הפה). מסננים את התמונות כך שישארו רק תמונות חזיתיות, כלומר, פנים שהזוית שלהם ביחס למרכז לא גדולה יותר מ-10°. מייצבים את התמונות ע״י הפעלה של טרנספורמציה אפינית על כל אחת מהן. טרנספורמציות

אלה מחושבות ע״י שימוש בשיטת הריבועים הפחותים (linear least squares, ראה 14.2.5) על נקודות ההשוואה ועל נקודות מתאימות שמסומנות על תבנית כלשהי של פנים.

8.2 חלוקה לאזורי פנים שונים

כל מאפיין מקומי מתוך קבוצת המאפיינים המקומיים שנתאר בהמשך, מתאר אזור פנים ספציפי. אזורי הפנים נוצרים מחלוקה של הפנים ל-10 אזורי פנים שונים המסומנים באותיות אנגלית גדולות (ראה איור 13). אזורי הפנים הם מספיק גדולים כך ששינויים קטנים בין פנים של אותו אדם לא ישפיעו עליהם במידה גדולה, ובנוסף, קיימת ביניהם חפיפה קטנה כדי שאם הייצוב של התמונות לא מדויק לגמרי, ימנע המצב בו מאפיינים יהיו מחוץ לאזור הפנים המתאים להם (למשל נניח שהעיניים בתמונה הן בגובה האזורים הירוקים אבל הן מוסטות מעט לכיוון ימין. אז יכול להיות שפיקסלים חשובים המרכיבים את העין השמאלית יהיו חלק גם מאזור ירוק וגם מאזור אדום. תהיה להם אולי תרומה מיותרת לאזור האדום אבל הם עדיין יהיו גם חלק מהמאפיינים של אזור העיניים הירוק).



איור 13 – חלוקה לאזורי פנים

אזורי הפנים המשמשים ליצירת המאפיינים המקומיים. משמאל, אזור הפנים המסמן את כל אזור הפנים, מימין חלקים שונים של הפנים. כל אזור פנים מסומן באות אנגלית גדולה.

8.3 קבוצת מאפייני הבסיס (low-level features)

המסווגים הנוצרים בשלב הסופי של הלמידה, מבוססים על קבוצה גדולה של מאפיינים מקומיים. קבוצה זו כוללת את המאפיינים הנוצרים באופן הבא: עבור כל אחד מאזורי הפנים שתוארו בסעיף הקודם, יוצרים את כל המאפיינים האפשריים המכילים קומבינציות שונות של המידע המתואר בשלושת הקטגוריות הבאות:

1. סוגים שונים של ערכי פיקסלים (pixel type):
   a. פורמט הצבע RGB (r)
   b. פורמט הצבע HSV (h)
   c. ערכים של גווני אפור Image intensity (i)

d. ערכי מגניטודות Edge magnitude (m) (ראה 10.1.1)

e. כיווני הגרדיאנטים Edge orientation (o) (ראה 10.1.1)

2. נורמליזציה (normalization) – כל מאפיין מתייחס לאזור הפנים המתאים לו לאחר אחת משתי שיטות הנורמליזציה הבאות: נורמליזציה של ממוצע (mean normalization) ונורמליזציה של אנרגיה (energy normalization). נורמליזציה של ממוצע מחלקת כל ערך בערך הממוצע ($\hat{x} = \frac{x}{\mu}$), וכאשר משתמשים בנורמליזציה של אנרגיה, מפחיתים מכל ערך את הממוצע ומחלקים את התוצאה בסטיית התקן ($\hat{x} = \frac{x-\mu}{\sigma}$).

3. ערכים גלובליים (aggregations) – היסטוגרמה של הערכים, או הממוצע וסטיית התקן שלהם.

לכל מאפיין מתאימים את השם שמגדיר אותו, המורכב משרשור תכונותיו:

"ערכים גלובליים:נורמליזציה:סוג הפיקסלים:אזור הפנים"

ערך כל תכונה היא האות האנגלית המתאימה לה, כמתואר בטבלה 1. ע"י שימוש בשיטה זו מתקבלת קבוצה גדולה של מאפיינים שמספרם הוא סך כל הקומבינציות האפשריות של הקטגוריות שהוצגו.

סוג הפיקסלים	נורמליזציה	ערכים גלובליים
RGB (r) HSV (h) ערכים של גווני אפור Intensity (i) ערכי מגניטודות Edge magnitude (m) כיווני גרדיאנטים Edge orientation (o)	ללא none (n) נורמליזציה של ממוצע Mean-normalization (m) נורמליזציה של אנרגיה Energy-normalization (e)	ללא none (n) היסטוגרמה histogram (h) סטטיסטיקה statistics (s)

טבלה 1 – סוגי הערכים המרכיבים את המאפיינים

בטבלה מוצגים כל סוגי הערכים והאותיות בהן משתמשים לסמנם. כל מאפיין מקבל את השם המורכב מהמבנה: "ערכים גלובליים:נורמליזציה:סוג הפיקסלים:אזור הפנים"
למשל "F:i:n:s" הוא המאפיין שמתאר את הערכים של גווני האפור של אזור המצח (ראה איור 13), ללא נורמליזציה ומוסיף גם המידע הכולל את הערך הממוצע וסטיית התקן שלהם.

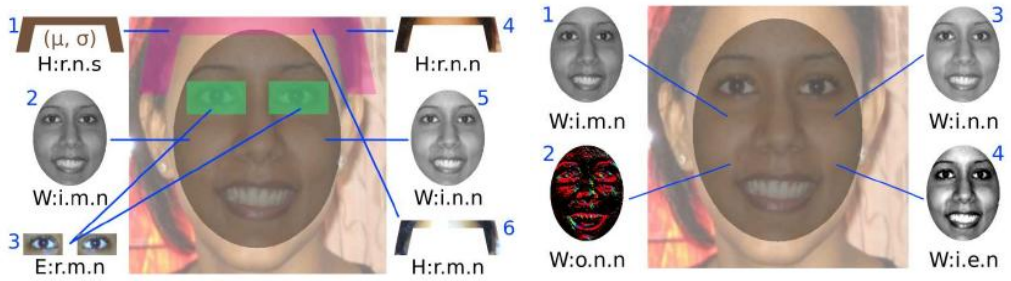
8.4 מסווגים של תיאורים (attribute classifiers)

בכדי ליצור מסווגים מתוך הקבוצה הגדולה של מאפייני הבסיס שנוצרה, משתמשים בשילוב של SVM עם גרסא של Adaboost (ראה 7.4, 7.5) הגרסא של אלגוריתם ה-Adaboost בה משתמשים

מתוארת ב-[59]. בשלב הראשון, יוצרים מכונת SVM עם פונקציית גרעין RBF עבור כל מאפיין בקבוצת המאפיינים שתוארה בסעיף הקודם SVM. לשם כך משתמשים בספרייה LibSVM [61] ספריות קוד פתוח לתמיכה בשימוש במכונות SVM. אותן מכונות SVM שנוצרו מהוות את קבוצת המסווגים החלשים שמתוכה בוחר Adaboost.

רוב זמן הריצה של אלגוריתם זה הוא סביב יצירת המסווגים החלשים ועדכון כל המשקלים עבור כל מאפיין בכל איטרציה. למרות שזהו זמן של תהליך הלמידה ולא של תהליך הריצה, קיים צורך לצמצם אותו. לכן משתמשים בשיטה זו בבחירת מאפיינים מראש (forward feature selection, ראה 7.4). כלומר המסווגים נוצרים פעם אחת בהתחלת הריצה, ובכל איטרציה בוחרים את המסווג אשר ביחד עם קבוצת המסווגים הנוכחית (מתחילים מקבוצה ריקה), נותן את השיפור הכי גדול.

בסיום הריצה של Adaboost עבור תכונה ספציפית, מתקבל מסווג חזק שמחליט במידת ודאות כלשהי אם התכונה נמצאת בתמונת קלט או לא. מסווג זה מבוסס על מספר מאפיינים קטן מתוך הקבוצה הגדולה של המאפיינים שתוארה בסעיף הקודם, מספר זה הוא בין 2 ל-6 לפי הצורך בדיוק מול מהירות ריצה סבירה. הדגמה של המאפיינים שנבחרו עבור התיאורים מגדר וצבע שיער נתונה באיור 14.



(ב) (א)

איור 14 – מאפיינים של מגדר וצבע שיער

הדגמה של המאפיינים שנבחרו עבור מסווגים של מגדר (א) וצבע שיער (ב) . מסביב לכל דוגמא מוצגים המאפיינים שעליהם המסווג הסופי החזק של תכונה זו מתבסס. כל מאפיין מצוין בשמו כפי שהוגדר ב- 8.3.

קיימות שתי בעיות בשימוש הנ״ל ב-Adaboost. הראשונה היא שכל מסווג שנוצר בנוי מכמה מכונות SVM, בסופו של דבר משתמשים בהרבה מסווגים כאלה ולכן יש לאחסן מספר גבוה של מכונות SVM בזיכרון, ולהריצן על כל תמונת קלט בתהליך הבחינה. השניה היא שכל מסווג של תכונה מוגבל בשימוש של צירוף לינארי בלבד של כמה מאפיינים שונים, בעוד שהאפשרות להשתמש בשילוב לא לינארי שלהם נחסמת.

תיקון לשתי בעיות אלה ניתן להשיג ע״י בניית מכונת SVM גלובאלית עבור כל תכונה המבוססת על המאפיינים שנבחרו עבור התכונה. כלומר, במקום לבנות את המסווג החזק לפי הנוסחא הקלאסית של Adaboost המצרפת באופן לינארי את המסווגים החלשים, מאחדים את כל המאפיינים שנבחרו עבור המסווג החזק, ובונים מסווג המבוסס על SVM על קבוצת מאפיינים זאת.

ניסויים בפועל הראו שמסווגים אלה משיגים אחוזי שגיאה נמוכים יותר בעוד שהם רצים מהר יותר ומצריכים פחות מקום בזיכרון [59].

8.5 מסווגים של דימויים (simile classifiers)

בניגוד למסווגים של תיאורים, מסווגים המבוססים על דימויים מוסיפים מידע על תכונות של תמונות אשר ההגדרה שלהן מתבססת על השוואה לתמונות אחרות של אנשים הנמצאים בקבוצת ייחוס נתונה. לדוגמא, במקום למצוא הגדרה מילולית לצורה כלשהי של עיניים, משתמשים במידת הדימיון של העיניים לכמה זוגות עיניים נתונות של אנשים מקבוצת הייחוס. גם חוסר דימיון יכול להיות יעיל לתיאור פנים, ניתן למשל לאפיין עיניים על ידי מידת הדימיון שלהם לעיניים של בראד פיט אבל גם ע"י מידת השוני שלהם מהעיניים של ריצ'רד גיר.

האנשים המופיעים בקבוצת הייחוס מתויגים בתג R_i, כאשר i הוא האינדקס שלהם בקבוצת הייחוס. אותם אנשים אינם מופיעים במאגר הנתונים LFW או בתמונות אחרות בהן משתמשים בתהליך הבחינה.

דימוי של אדם R_i הוא בחירה של אזור פנים מתוך קבוצה של 8 אזורי פנים שנבחרו מתוך סך כל אזורי הפנים שתוארו בסעיף 8.2, ביניהם למשל פה, גבות, אף ועיניים. עבור כל דימוי, כלומר, עבור כל אדם R_i וכל אחד מאזורי הפנים, יוצרים מסווגים חלשים המתבססים על מאפייני הבסיס אשר תוארו בסעיף 8.3. באופן דומה לבניית המסווגים של תיאורים, משתמשים ב-Adaboost בכדי לבנות מסווג חזק המבוסס על כמה מסווגים חלשים (בניסויים שבודקים דימויים נבחרו 6 מסווגים) עבור כל דימוי. לדוגמא, אחד מהמסווגים החזקים הנוצרים הוא מסווג שקובע עבור פני קלט נתונים האם העיניים בתמונת הקלט דומות לעיניים בתמונת הפנים R_1.

קבוצת הלמידה של כל מסווג על אדם R_i מקבוצת הייחוס מורכבת מקבוצת תמונות חיוביות של עד 600 תמונות של אותו אדם ועד 6000 תמונות שליליות שהן תמונות של אדם אחר. באיור 15 ניתן לראות דוגמאות חיוביות ושליליות לבניית מסווגים לדימויים שונים.

8.6 תהליך הזיהוי

כפי שציינו קודם, שיטה זו מתארת את בנייתם של שני מסווגים – אחד המבוסס על תיאורים ושני המבוסס על דימויים. בניית מסווגים אלה נעשית באופן דומה עבור שניהם:

1. בוחרים k מאפייני בסיס בהם נשתמש כאשר k מספר כלשהו, גדול מספיק בשביל להגיע לאחוזי הדיוק הרצויים. עבור כל תמונת קלט I, מרכיבים מהמאפיינים שנבחרו $f_{i=1\dots k}$ את הוקטור: $F(I) = \langle f_1(I), \dots, f_k(I) \rangle$.
2. עבור כל אחד מהוקטורים שחושבו בצעד הראשון, בונים מסווג עבור n הדימויים או התיאורים $C_{i=1\dots n}$ ומרכיבים את הוקטור: $C(I) = \langle C_1(F(I)), \dots, C_n(F(I)) \rangle$.

3. בהינתן שתי תמונות פנים I_1, I_2, נסמן את וקטורי המסווגים על כל אחת מהתמונות: $a_i = C_i(I_1)$ ו-$b_i = C_i(I_2)$. עבור כל אחד מ-n המסווגים מחשבים את זוג הערכים:

$$p_i = \left(|a_i - b_i| \cdot g(\tfrac{1}{2}(a_i + b_i)), (a_i \cdot b_i) \cdot g(\tfrac{1}{2}(a_i + b_i))\right) \quad (8)$$

כאשר g היא התפלגות נורמלית סטנדרטית (גאוסיאן, ראה 14.2.2).

איור 15 – למידת מסווגים של דימויים

קבוצת הלמידה ליצירת מסווגים של דימויים מורכבת מדוגמאות שליליות וחיוביות עבור אזור פנים של אדם כלשהו מקבוצת הייחוס. משמאל ניתן לראות 3 דוגמאות חיוביות ובימין 3 דוגמאות שליליות עבור 4 אזורי פנים של שני אנשים מקבוצת הייחוס.

משרשרים את כל הזוגות האלה לוקטור בן 2n מימדים וכעת מפעילים מסווג המכריע האם וקטור זה מייצג שתי תמונות של אותו אדם או לא. מסווג זה הוא מכונת SVM עם פונקציית ליבה RBF והוא עובר תהליך למידה מוקדם כשקבוצת הלמידה שלו מכילה זוגות חיוביים (פנים של אותו אדם) וזוגות שליליים (פנים של אנשים שונים).

9 שיטות למידה מטריציוניות

9.1 רקע

אלגוריתמי סיווג המשתמשים במרחקים אוקלידים כגון kNN המשתמש במרחקים בשביל למצוא את קבוצת השכנים של נקודה נתונה, שימוש במתארי SIFT שבו מודדים מרחקים בין מתארי SIFT של תמונות ביניהן משווים וכדומה, אינם מביאים תמיד לתוצאות מספקות לאור העובדה כי המרחבים הנפרשים ע״י תמונות פנים הם לרוב מרחבים שאינם לינאריים ולכן שימוש במרחקים אוקלידיים אינו מאפיין את המרחקים האמיתיים לצורך מטרות ההשוואה. מרחקים אוקלידיים אינם מתחשבים בסוג המידע המבוקש, ומרחקים בין נקודות המייצגות פנים יכולים להיות שונים, כתלות במטרת הסיווג המשליכה צורה שונה של מרחב. לדוגמא, סיווג מגדרי לעומת סיווג לפי גיל יכול להכתיב מרחקים שונים בין הדוגמאות. על-מנת להתמודד עם בעיות אלה זו משתמשים במרחקי מהלנוביס. מרחקים אלה מחושבים כמרחקים אוקלידים על הנתונים, לאחר שעברו טרנספורמציה לינארית (ראה 14.1.3) ולכן אפשר באמצעותם לשלוט על אופי התנהגותם של המרחקים בין דוגמאות הפנים. בשיטות למידה מטריציונית (metric learning) מנסים ללמוד מטריצה המבטאת מרחקי מהלנוביס שמאפיינת הכי טוב את הקשרים בין הדוגמאות ע״י תהליך של למידה וכך לומדים את ערכי המרחקים המתאימים ביותר לקשרים בין הדוגמאות. הלמידה בדרך כלל מכוונת ע״י פונקציית מטרה אשר משפיעה של אופיים של המרחקים הנוצרים. השוני המהותי, אם-כן, בין השיטות הוא בדרך כלל בפונקציית המטרה.

סעיף זה מתאר שלוש שיטות של למידה מטריציוניות ושיטה המשתמשת ב- kNN בשביל זיווג פנים ובלמידה מטריציונית בשביל לשפר את ביצועיו של ה-kNN [62]. נתאר תחילה שתי שיטות ידועות ושימושיות: הראשונה היא מטריצות שכנים קרובים ביותר עם שוליים רחבים (Large Margin Nearest Neighbor Metrics – LMNN) והשניה היא שיטת למידה מטריציונית באמצעות שיטות מתורת האינפורמציה (Information-Theoretic Metric Learning- ITML). לאחר מכן מתוארות שתי שיטות חדשניות המשיגות תוצאות מבין הטובות הקיימות היום על מאגר הנתונים LFW בגישה הבלתי מוגבלת (ראה 6.5). הראשונה היא למידה מטריציונית מבוססת הפרדה לוגיסטית (Logistic Discriminant based Metric Learning – LDML) והשניה היא k השכנים הקרובים ביותר עם שוליים (Marginalized kNN – MkNN).

בשיטות אלה תמונה מיוצגת כוקטור רב מימדים, ובשביל להגיע סיבוכיות סבירה מפעילים תחילה את אלגוריתם ה-PCA (ראה 4.2) להורדת מימדים.

9.2 מרחקי מהלנוביס

ניזכר בהגדרה של מרחק אוקלידי ריבועי:

$$d(x_i, x_j) = \|(x_i - x_j)\|^2 = (x_i - x_j)^\top (x_i - x_j) \qquad (9)$$

בהינתן מטריצה $\boldsymbol{L} \in \mathbb{R}^{DxD}$ המבטאת טרנספורמציה לינארית (ראה 14.1.3), מרחק מהלנוביס (Mahalanobis distance) בין $x_i, x_j \in \mathbb{R}^D$ הוא:

$$d_{\boldsymbol{L}}(x_i, x_j) = \|\boldsymbol{L}(x_i) - \boldsymbol{L}(x_j)\|^2 = \|\boldsymbol{L}(x_i - x_j)\|^2 \qquad (10)$$

כלומר הוא המרחק האוקלידי הריבועי לאחר הפעלתה של הטרנספורמציה הלינארית $\boldsymbol{L}$.

מאחר ו-

$$d_{\boldsymbol{M}}(x_i, x_j) = \|\boldsymbol{L}(x_i - x_j)\|^2 = \qquad (11)$$
$$\left(\boldsymbol{L}(x_i - x_j)\right)^\top \boldsymbol{L}(x_i - x_j) = (x_i - x_j)^\top \boldsymbol{L}^\top \boldsymbol{L}(x_i - x_j)$$

מרחק זה יכול להיות מבוטא גם באופן הבא:

$$d_{\boldsymbol{M}}(x_i, x_j) = (x_i - x_j)^\top \boldsymbol{M}(x_i - x_j) \qquad (12)$$

כאשר $\boldsymbol{M} \in \mathbb{R}^{DxD}$ היא המטריצה $\boldsymbol{L}^\top \boldsymbol{L}$. $\boldsymbol{M}$ נקראת מטריצת מהלנוביס (Mahalanobis metric). על-מנת לקבל תמיד מרחקים לא שליליים, $\boldsymbol{M}$ היא מטריצה סימטרית וחיובית לחלוטין (ראה 14.1.1).

9.3 מטריצות שכנים קרובים ביותר עם שוליים רחבים (Large Margin Nearest Neighbor Metrics – LMNN)

שיטה זו [63] פותחה לפני כמספר שנים בניסיון לשפר את האלגוריתם kNN. המטרה היא ללמוד את המרחקים הטובים ביותר למטרת חישוב סיווג. מטרה זו מושגת ע"י למידה של טרנספורמציה לינארית השואפת להקטין מרחקים בין נתונים עם אותו סיווג ולהגדיל מרחקים בין נתונים בעלי סיווג שונה. במילים אחרות, השאיפה של הטרנספורמציה הלינארית היא להעביר את הנתונים למרחב בו עבור כל נתון, k השכנים הכי קרובים אליו מאותו תג, הם מאוד קרובים אליו, וכל נתון בעל תג שונה מספיק רחוק ממנו.

לשם כך מגדירים את המונחים:

- שכני מטרה (target neighbors) – עבור כל וקטור x_i בקבוצת הלמידה מגדירים את k הוקטורים שנרצה שיחשבו כ-k שכנים של x_i לאחר הפעלתה של הטרנספורמציה הלינארית משתמשים בסימון $j \rightsquigarrow i$ בשביל לציין כי x_i הוא שכן מטרה של x_i. נשים לב שאם $j \rightsquigarrow i$ אז לאו דווקא מתקיים $i \rightsquigarrow j$.

- מתחזים (impostors) - וקטורים הקרובים ל-x_i לפחות כמו אחד מ-k שכני המטרה שלו אבל הם לא מסווגים באופן זה.

עבור כל וקטור מגדירים את הרדיוס בו שוכנים שכני המטרה שלו וכל וקטור עם סיווג אחר השוכן ברדיוס זה הוא מתחזה. המטרה היא לקרב כמה שאפשר את שכני המטרה אל הוקטור שעבורו הם מוגדרים ולהקטין כמה שאפשר את מספר המתחזים. בכדי להגדיל את מידת העמידות לרעשים מוסיפים עוד שוליים לרדיוס זה והשאיפה היא להרחיק אף יותר (מעבר לשוליים) וקטורים עם סיווג שונה. מכאן שמה של השיטה - שכנים קרובים ביותר עם שוליים רחבים. באופן פורמלי, עבור וקטור קלט x_i עם תג y_i, יהי x_j שכן מטרה כלשהו. מתחזה הוא כל וקטור x_l עם תג שונה מ-y_i המקיים:

$$\|\boldsymbol{L}(x_i - x_l)\|^2 \leq \left\|\boldsymbol{L}(x_i - x_j)\right\|^2 + 1 \qquad (13)$$

ההגדרה המקובלת אם-כן, לשוליים הוא גודל יחידה אך כמובן ניתן להגדיר גודל שונה עבורם.

בשביל להשיג את המטרות שתוארו, מגדירים פונקציית הפסד (loss function) בעלת שני חלקים, חלק אחד אחראי על הקטנת מרחקים בין נתונים עם סיווג זהה, חלק שני אחראי על הגדלת מרחקים בין נתונים עם סיווג שונה.

נגדיר באופן פורמלי את שני חלקיה של פונקציית ההפסד. החלק הראשון מוגדר באמצעות סכום המרחקים הריבועיים לפי הטרנספורמציה הלינארית $\boldsymbol{L}$ בין כל שכני המטרה הוא:

$$\varepsilon_{pull}(\boldsymbol{L}) = \sum_{j \rightsquigarrow i} \left\|\boldsymbol{L}(x_i - x_j)\right\|^2 \qquad (14)$$

שואפים להקטין את הסכום הזה על מנת למשוך (to pull) שכני מטרה אחד אל השני.

החלק השני מוגדר באמצעות התייחסות לזוגות של נתונים המסווגים באופן שונה אבל שבכל זאת לפחות אחד מהם קרוב לשני יותר מאשר שכני המטרה שלו (זוגות המכילים מתחזה). סוכמים את המרחקים שיש להוסיף לזוגות כאלה בכדי שיהיו מספיק רחוקים אחד מהשני (לפחות כמו שכני מטרה, פלוס מרחק יחידה שמציין את השוליים). לשם הגדרת סכום זה נגדיר את הערך y_{il}:

$$y_{il} = \begin{cases} 1 & y_i = y_l \\ 0 & \text{אחרת} \end{cases}$$

והסכום המתאר את החלק השני של פונקציית ההפסד הוא:

$$\varepsilon_{push}(\boldsymbol{L}) = \sum_{i,j \rightsquigarrow i} \sum_l (1 - y_{il}) \left[1 + \left\|\boldsymbol{L}(x_i - x_j)\right\|^2 - \|\boldsymbol{L}(x_i - x_l)\|^2\right]_+ \qquad (15)$$

כאשר: $[z]_+ = max\ (z, o)$. בכדי להבין את האינטואיציה מאחורי סכום זה, ניזכר בהגדרה של מתחזה ונשים לב שהמרחק $1 + \left\|\boldsymbol{L}(x_i - x_j)\right\|^2 - \|\boldsymbol{L}(x_i - x_l)\|^2$, כאשר x_i ו-x_l הם נתונים בעלי סיווג שונה ו-x_j הוא כל שכן מטרה אפשרי של x_i, יהיה שלילי אם $1 + \left\|\boldsymbol{L}(x_i - x_j)\right\|^2 \leq \|\boldsymbol{L}(x_i - x_l)\|^2$, כלומר אם x_l הוא לא מתחזה. באופן דומה, הוא יהיה חיובי אם הוא כן מתחזה. לכן בעצם הסכום $\varepsilon_{push}(\boldsymbol{L})$ מכיל את כל המרחקים שיש להוסיף, או לדחוף (to push) אל הנתונים המתחזים, על מנת להרחיקם מספיק מנתונים כך שהם לא יהיו מתחזים.

לבסוף, פונקציית ההפסד משלבת את שני החלקים הנ"ל באופן הבא:

$$\varepsilon(\boldsymbol{L}) = (1 - \mu)\varepsilon_{pull}(\boldsymbol{L}) + \mu\varepsilon_{push}(\boldsymbol{L}) \qquad (16)$$

כאשר $\mu \in [0,1]$ הוא משקל המאפשר לשלוט על האיזון בין המשקל שניתן לקירוב שכנים והרחקת המתחזים, מאחר ושתי פעולות אלה במקרים מסוימים מתנגשות. מניסויים שנעשו, בחירה שונה של הערך μ אינה משנה באופן משמעותי את הפתרון והערך $\mu = \frac{1}{2}$ עובד באופן יעיל יחסית. בהינתן פונקציית ההפסד $\varepsilon(\boldsymbol{L})$, ניתן להמיר את בעיית הבאתה למינימום לבעייה דומה מסוג תכנות מוגדר- חלקית (semidefinite program – SDP) אשר קיים אלגוריתם ידוע לפתרונה ([63]). באמצעות פתרון זה לומדים את הטרנספורמציה $\boldsymbol{L}$ אשר מביאה למינימום את $\varepsilon(\boldsymbol{L})$ והיא קובעת את מטריצת המרחקים $\boldsymbol{M} = \boldsymbol{L}^{\mathsf{T}}\boldsymbol{L}$.

9.4 למידה מטריציונית באמצעות שיטות מתורת האינפורמציה (Information-Theoretic Metric Learning)

עוד שיטה שלומדת מטריצת מהלנוביס היא ITML [64] ומתוארת בסעיף זה. ניזכר תחילה בהגדרתו של מרחק מהלנוביס שחישבנו בנוסחה (12):

$$d_{\boldsymbol{A}}(x_i, x_j) = (x_i - x_j)^{\mathsf{T}}\boldsymbol{M}(x_i - x_j) \qquad (17)$$

בתהליך הלמידה, אם-כן, מתקבלת מטריצה $\boldsymbol{M}$ חיובית לחלוטין המגדירה את מרחקי המהלנוביס בין תמונות שונות.

מניחים שנתון ידע מסוים לגבי המרחקים בין הדוגמאות (יודעים לפחות אם תמונות הן באותה מחלקת פנים או לא). מגדירים מדד של דימיון או חוסר דימיון באופן הבא: שתי דוגמאות הן דומות אם מרחק המהלנוביס ביניהן קטן מחסם עליון נתון כלשהו: $d_{\boldsymbol{M}}(x_i, x_j) \leq u$, והן שונות אם מרחק המהלנוביס שלהן גדול מחסם תחתון נתון כלשהו: $d_{\boldsymbol{M}}(x_i, x_j) \geq \ell$ אי שיוונות אלה, עבור כל זוג בקבוצת הלמידה, מהווים את אילוצי המרחק של המערכת.

אנו מניחים שנתונה פונקציית מרחקי מהלנוביס $\boldsymbol{M}_0$ ידועה המסתמכת על מידע מוקדם ידוע. בהרבה מקרים, יש ידע מוקדם על פונקציית מרחקי מהלנוביס או על התפלגות הנתונים המשפיעה על פונקציה זאת (למשל אם התפלגות הנתונים גאוסיאנית, מטריצת מרחקים מתאימה היא המטריצה ההופכית למטריצת השונות). נרצה להביא את המטריצה $\boldsymbol{M}$ להיות קרובה ככל האפשר למטריצה $\boldsymbol{M}_0$, תוך כדי שמירה על אילוצי המרחק. לפי תיאוריות של תורת האינפורמציה קיימת פונקציה על מקבוצת פונקציות מרחקי מהלנוביס אל קבוצה של גאוסיאנים בעלי אותו ממוצע ושונות משתנה. לכן עבור כל מטריצת מרחק מהלנוביס, $\boldsymbol{M}$, ניתן להתייחס להתפלגות הגאוסיאנית המתאימה לה:

$$p(x;\boldsymbol{M}) = \frac{1}{z} exp\left(-\frac{1}{2} d_A(x,\mu)\right) \qquad (18)$$

כאשר z הוא מספר המשמש לנירמול, ו-$\boldsymbol{M}$ היא המטריצה ההופכית למטריצה השונות של ההתפלגות.

כך ניתן למדוד את המרחק בין שתי פונקציות המרחק המיוצגות ע"י $\boldsymbol{M}$ ו-$\boldsymbol{M}_0$ ע"י האנטרופיה היחסית (ראה 14.2.4) של הגאוסיאנים המתאימים להם:

$$KL(p(x;\boldsymbol{M}_0)||p(x;\boldsymbol{M})) = \int p(x;\boldsymbol{M}_0) log \frac{p(x;\boldsymbol{M}_0)}{p(x;\boldsymbol{M})} dx \qquad (19)$$

ואם נגדיר את S כקבוצת זוגות הנקודות המייצגות פנים של אותו אדם ואת D כקבוצת זוגות הנקודות המייצגות פנים של אנשים שונים, ניתן לנסח את הבעיה באופן הבא:

$$min_{\boldsymbol{M}} \quad KL(p(x;\boldsymbol{M}_0)||p(x;\boldsymbol{M}))$$

$$s.t. \quad d_{\boldsymbol{M}}(x_i, x_j) \leq u \qquad (i,j) \in S$$

$$d_{\boldsymbol{M}}(x_i, x_j) \geq \ell \qquad (i,j) \in D$$

באלגוריתם לפתרון הבעיה, המשתמש בהצגה שלה כבעיית אופטימיזציית ברגמן (Bregman optimization problem) [64] משתמשים בפרמטר נוסף, γ, שקובע את האיזון בין השאיפה להתקרב למטריצה $\boldsymbol{M}_0$ ובין השאיפה לעמוד באילוצים. אם-כן, הפרמטרים הנתונים בתחילת האלגוריתם ללמידת מטריצת המהלנוביס $\boldsymbol{M}$ הם $\boldsymbol{M}_0, u, \ell$ ו-γ.

9.5 למידה מטריציונית מבוססת הפרדה לוגיסטית (Logistic Discriminant based Metric Learning)

שיטה זו מתבססת על הערכות הסתברותיות המנסות לחזות האם שתי תמונות פנים מכילות פנים של אותו אדם. קובעים את ההסתברות p_k שזוג תמונות $k = (i, j)$ הוא זוג חיובי (פנים של אותו אדם), כלומר שהתג שלו t_k הוא 1, באופן הבא:

$$p_k = p\big(y_i = y_j | x_i, x_j; M, b\big) = \sigma\left(b - d_M\big(x_i, x_j\big)\right) \qquad (20)$$

כאשר $\sigma(z) = (1 + exp\ (-z))^{-1}$ הוא פונקציית סיגמואיד (sigmoid, ראה 14.2.6) ו-b הוא ניטאי (bias term). מטרת תהליך הלמידה היא להגיע ל-$\boldsymbol{M}$ ול-b.

את האינטואיציה מאחורי שימוש בפונקציית סיגמואיד ניתן להבין אם נשים לב שהקלט של המודל הוא מרחק בין זוג נקודות המייצגות פנים, והפלט הוא ההסתברות של אותו זוג לייצג פנים של אותו אדם. פונקציית הסיגמואיד מתאימה מאחר שהיא מחזירה הסתברות נמוכה עבור ערכים קטנים (במקרה שלנו מרחקים גדולים) ובאופן יחסית חד עוברת להחזיר הסתברויות גבוהות עבור הערכים הגדולים (מרחקים קטנים). היתרון בשימוש בסיגמואיד על פני פונקציית הכרעה בינארית המשתמשת בערך סף הוא שהיא מאפשרת התנהגות רציפה במעבר בין קבוצות ההערכה (אותו אדם או לא).

$d_{\boldsymbol{M}}\big(x_i, x_j\big)$ הוא צירוף לינארי של הערכים המופיעים במטריצה $\boldsymbol{M}$ כלומר ניתן להגיע ע"י פיתוח סכום זה לצורה:

$$d_{\boldsymbol{M}}\big(x_i, x_j\big) = \big(x_i - x_j\big)^{\mathsf{T}} \boldsymbol{M} \big(x_i - x_j\big) = X_1 m_{11} + \qquad (21)$$
$$X_2 m_{12} + \cdots + X_{n\times n} m_{nn}$$

כאשר X_k הוא וקטור בגודל $n \times n$ המכיל את הערכים של $\big(x_i - x_j\big)^{\mathsf{T}}\big(x_i - x_j\big)$ עבור הזוג $k = (i, j)$. אם נסמן ב-W את הוקטור בגודל $n \times n$ המכיל את הערכים של $\boldsymbol{M}$, אזי ניתן לבטא את p_k באופן הבא:

$$p_k = \sigma(b - W^{\mathsf{T}} X_k) \qquad (22)$$

זהו מודל הבחנה לוגיסטי לינארי סטנדרטי (פונקציית ההבחנה תלויה באופן לינארי בפרמטרים). משתמשים בשיטת הנראות המקסימלית על מנת למצוא את הערכים האופטימליים של $\theta = (b, W^{\mathsf{T}})$ (ראה 14.2.7). נגדיר פונקציית נראות שמתארת את ההסתברות של כל הזוגות החיוביים בקבוצת הלמידה להיות חיוביים ושל כל הזוגות השליליים להיות שליליים בהינתן הפרמטרים $\theta = (b, W^{\mathsf{T}})$. כלומר אם נסמן ב-$c_k$ את המאורע שזוג k מקבל תג נכון, אזי רוצים

למצוא θ שממקסם את ההסתברות $P(c_1, c_2, \dots, c_{n\times n}|\theta)$. מאחר והמודל ההסתברותי שהגדרנו ב- (20) מניח אי תלות בין הזוגות, הסתברות זו שקולה להסתברות:

$$\prod_{k=1}^{n\times n} P(c_k|\theta) \tag{23}$$

מפעילים על פונקציה זו ln בשביל להשתמש בסכום במקום מכפלה ומקבלים:

$$ln \prod_{k=1}^{n\times n} P(c_k|\theta) = \sum_{k=1}^{n\times n} ln\big(P(c_k|\theta)\big) \tag{24}$$

צריך למצוא את הערכים של θ שממקסמים את הסכום מהצד הימין של השיוויון. נשים לב שסכום זה הוא בעצם:

$$\mathcal{L} = \sum_{k=1}^{n\times n} t_k ln p_k + (1 - t_k)\, ln(1 - p_k) \tag{25}$$

משום שעבור כל זוג, אם הוא חיובי אז $t_k = 1$ ואז הוא תורם לסכום את ההסתברות שלו להיות חיובי p_k. אם הוא שלילי אז $1 - t_k = 1$ ואז הוא תורם לסכום את ההסתברות שלו להיות שלילי $1 - p_k$. הגרדיאנט של הסכום הוא:

$$\nabla\mathcal{L} = \sum_k (t_k - p_k)\, X_k \tag{26}$$

אלגוריתם האופטימיזציה שמשתמשים בו הוא עלייה בגרדיאנט (gradient ascent). אלגוריתם אופטימיזציה זה מנסה למצוא מקסימום מקומי ע"י לקיחת צעדים בגודל פרופורציונאלי לגרדיאנט הפונקציה. האלגוריתם מבוסס על הרעיון שאם הפונקציה $f(x)$ מוגדרת וגזירה ב-a, אז $f(x)$ עולה מהר יותר אם "מטיילים" מ-a בכיוון הגרדיאנט החיובי $b = a + \gamma\nabla f(a)$. אם γ הוא מספר חיובי מספיק קטן, אזי $f(b) \geq f(a)$. השאיפה היא לבצע מספר צעדים כאלה עד להתכנסות וכך למצוא את המקסימום המקומי של $f(x)$.

מאחר והפונקציה $\mathcal{L}$ היא פונקציה קעורה וחלקה, מציאת המקסימום שלה היא משימה פשוטה ומהירה – דבר המקנה יתרון של שיטה זו לעומת השיטות הקודמות שתוארו קודם, ITML ו-LMNN.

9.6 kNN עם שוליים - Marginalized MkNN (MkNN)

שימוש במטריצת מהלנוביס מעתיק את המרחב המקורי באמצעות טרנספורמציה לינארית למרחב חדש. מגבלה זו מקשה על הפרדה בין זוגות פנים חיוביים לזוגות פנים שליליים משום שמרחקים בין תמונות שונות של אותם פנים יכולים להיות לא לינאריים וגדולים ממרחקים בין תמונות של אנשים שונים עם אותה זווית פנים או הבעה. בפרק זה נציג דרך להשתמש ב-kNN עבור זיווג פנים. המסווג בו משתמשים הוא מסווג לא לינארי המסווג זוגות פנים באופן בינארי (פנים של אותו אדם או פנים של אדם שונה).

כזכור אלגוריתם kNN מצביע, עבור כל נקודה, על k שכנים הכי קרובים אליה ומחליט על סיווגה בהתאם לסיווג הרווח בקרב קבוצה זו.

נניח שנתונה נקודה כלשהי x_i במרחב הדוגמאות ו-n_c^i הוא מספר השכנים מתוך k השכנים הכי קרובים ל-x_i שהתג שלהם הוא c. מגדירים את ההסתברות ש-x_i תקבל תג c:

$$p(y_i = c|x_i) = {}^{n_c^i}/_{k} \qquad (27)$$

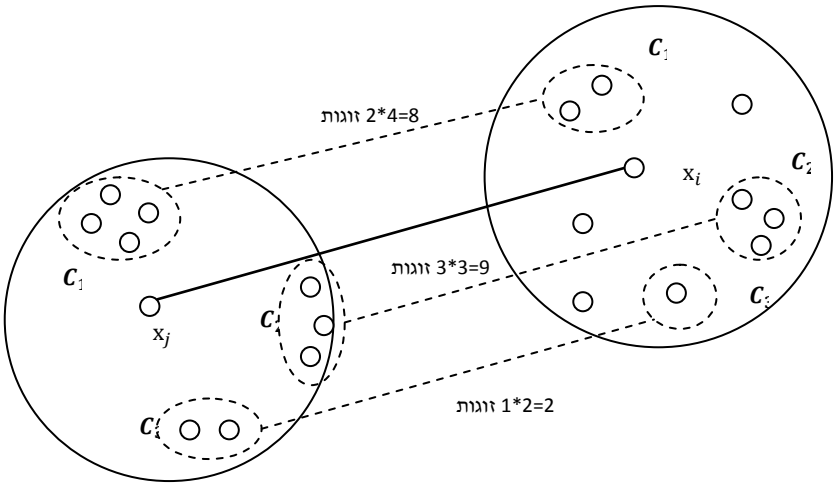
בהקשר זה התג c הוא שם של אדם ספציפי. באותו אופן, ההסתברות של זוג נקודות במרחב הדוגמאות, (x_i, x_j) לקבל שתיהן את התג c היא:

$$p(y_i = y_j = c|x_i, x_j) = k^{-2} n_c^i n_c^j \qquad (28)$$

וזאת היא בעצם ההסתברות של זוג לקבל תג זהה כלשהו, כלומר שאיבריו יהיו מסווגים לאותה מחלקה (זוג פנים של אותו אדם). הבעיה בה אנו מתעניינים היא ההכרעה האם זוג נקודות מקבל תג זהה או שונה, ולשם כך נתייחס לבעיה חדשה המוגדרת באופן הבא: נתייחס רק לזוגות של נקודות – כלומר אם קבוצת הלמידה המקורית היא בגודל n, כעת קבוצת הלמידה שלנו היא בגודל n^2. קבוצת השכנים הקרובים ביותר של זוג (x_i, x_j), היא בעלת k^2 איברים ומכילה את כל הקומבינציות האפשריות של זוגות שהאיבר הראשון שלהם שייך לקבוצת k השכנים הקרובים ביותר ל-x_i בבעיה המקורית והאיבר השני שלהם שייך לקבוצת k השכנים הכי קרובים ל-x_j. זוג (x_i, x_j) מסווג באופן בינארי בתג 0 או 1. הוא מקבל 1 אם התג של x_i ו-x_j זהה ו-0 אם הוא שונה.

חישבנו במשוואה (28) את ההסתברות של זוג דוגמאות (x_i, x_j) לקבל תג זהה וזאת היא בעצם ההסתברות שזוג זה יקבל תג 1 בבעיה החדשה. הסתברות זו היא מספר השכנים של הזוג (x_i, x_j) מבין קבוצת k^2 השכנים הקרובים ביותר אשר מתויגים ב-1, חלקי מספר השכנים הקרובים ביותר k^2. (ראה דוגמא באיור 16).

באופן זה מצטמצמים לבעיית kNN אחת שניתן לשפרה באמצעות אלגוריתם LMNN שהוזכר בסעיף 9.3 על מנת לחשב את המרחקים בין הנקודות ולקבוע ע"י מיהם השכנים של כל נקודה.



איור 16 – קביעת תג עבור זוג דוגמאות

דוגמא עבור k=10 עבור זוג דוגמאות נתון (x_i, x_j). מתייחסים ל10*10=100 השכנים הקרובים ביותר ל-(x_i, x_j) הנוצרים מכל הקומבינציות של השכנים הקרובים ביותר של x_i ו-x_j. מבין זוגות אלה ישנם 8+9+2=19 זוגות המסווגים באותו תג בבעיה המקורית ולכן הם מתויגים 1 בבעיה החדשה.

10 תבניות של מגניטודות מכוונות (POEM – Patterns of Oriented Edge Magnitudes)

השיטה המתוארת בסעיף זה היא שיטה חדשה המשיגה תוצאות מהטובות ביותר על מאגר הנתונים LFW בגישה המוגבלת (ראה 6.5). בנוסף לכך, הסיבוכיות של יצירת המתאר (descriptor) בה היא משתמשת מאוד נמוכה יחסית, למשל פי 20 מהצעד הראשון שנעשה בשיטות המבוססות על פילטרי Gabor [37].

השיטה משתמשת בשילוב של שתי שיטות ידועות - מאפייני LBP (ראה 5.3) והיסטוגרמות של גרדיאנטים (histograms of oriented gradients – HOG). הרעיון הכללי הוא שבמקום להשתמש בערכי הפיקסלים בחישוב מאפייני ה-LBP, משתמשים בהיסטוגרמות של מגניטודות (magnitudes) שהן גדלים של גרדיאנטים (gradients), כלומר במידע על דפוסי ההתנהגות של כיווני קווי המתאר (edges) שבתמונה. וליתר דיוק, מאפייני ה-LBP מחושבים כאשר ערכי הפיקסלים הם מאפייני ה-HOG שחושבו עבורם. מאחר ומאפייני HOG הם וקטורים, מאפייני ה-LBP מחושבים עבור כל מימד, ומיוצגים לבסוף גם כוקטורים בעלי אותו מספר מימדים.

10.1 רקע

10.1.1 תמונת הגרדיאנט (image gradient)

נתייחס לתמונה כאל פונקציה בעלת שני משתנים, $f(x,y)$. התחום שלה הוא קבוצת כל זוגות הקוארדינטות והטווח שלה הוא קבוצת ערכי הפיקסלים. עבור כל פיקסל, הוקטור הדו-מימדי המורכב מקצב שינוי ערכי הפיקסלים בכל קוארדינטה נקרא גרדיאנט (gradient) ומוגדר כדלהלן:

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \tag{29}$$

וקטור זה מצביע לכיוון של השינוי הגדול ביותר של f בנקודה (x,y). המגניטודה (magnitude) של הוקטור ∇f היא:

$$M(x,y) = |\nabla f| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} \tag{30}$$

ערך זה הוא הערך של קצב השינוי בכיוון הגרדיאנט. $M(x,y)$ היא פונקציה המתאימה לכל קואורדינטה של התמונה, (x,y), ערך חדש ולתמונה הנוצרת על-ידה קוראים תמונת הגרדיאנט (gradient image). דוגמא לתמונת גרדיאנט באיור 17.

תמונת הגרדיאנט היא כלי שימושי עבור מציאת קווי מתאר (egde detection) של תמונות מאחר ובאזורים אלה השינוי בערכי הפיקסלים בתמונה הוא שינוי גדול ביחס לשינוי בערכי הקואורדינטות.

איור 17 – תמונת גרדיאנט

דוגמא של תמונת הגדריאנט של תמונת פנים. ערך כל פיקסל בתמונה הוא ערך המגניטודה של הגרדיאנט המחושב עבור פיקסל זה.

10.1.2 היסטוגרמה של גרדיאנטים מכוונים (Histogram of oriented gradients - HOG)

שיטה זו [65] היא שיטה הפותחה עבור זיהוי אובייקטים בתמונה ומהווה השראה לשיטה המשתמשת במאפייני POEM המתוארת בסעיף זה. השימוש במתארי HOG מתבסס על הרעיון שניתן לתאר את צורתו של אובייקט באזורים מקומיים בתמונה ע״י התפלגותם של וקטורי הגרדיאנט בכל אזור מקומי. לצורך זה, מחלקים את התמונה לאזורים מקומיים קטנים הנקראים תאים (cells). עבור כל תא, מחשבים תחילה את תמונת הגרדיאנט של הפיקסלים שנמצאים בו, כלומר עבור כל פיקסל מחשבים את הגרדיאנט המתאים לו. אם התמונה היא צבעונית, מחשבים את תמונת הגרדיאנט עבור כל ערוץ של צבע. הגרדיאנט בעל המגניטודה הגדולה ביותר משמש כגרדיאנט של הפיקסל.

עבור כל תא, מכינים היסטוגרמה של כיווני הגרדיאנטים, כאשר המגניטודות (או פונקציות של המגניטודות) שלהם מהוות משקלים. כלומר, כל מחלקה של ההיסטוגרמה היא טווח כלשהו של מעלות, כאשר גרדיאנט השייך למחלקה כלשהו (המעלות שלו שייכות לתחום המעלות של אותה מחלקה), תורם למחלקה זאת ערך לפי המגניטודה שלו. האינטואיציה היא ששינויים חזקים ומהירים יותר בערכי הפיקסלים (מגניטודות גבוהות) נחשבים יותר.

נורמליזציה מתבצעת ע״י חלוקה של התמונה לחלקים גדולים יותר הנקראים בלוקים (blocks). לכל בלוק מסכמים מידה הנקראת אנרגיה (energy) עבור כל ההיסטוגרמות של התאים הנמצאים באותו בלוק ומשתמשים בסכום זה לנרמל את כל התאים בבלוק. בלוקים יכולים להיות מלבניים או מעוגלים, ויש כמה שיטות נורמליזציה שונות בהן ניתן להשתמש. עבור כל

בלוק, הוקטור המייצג את איחוד ההיסטוגרמות של הבלוק נקרא מתאר HOG (HOG descriptor – Histogram of Oriented Gradient descriptor).

לצורך זיהוי אובייקטים בתמונה, מתבצע מעבר על התמונה באמצעות חלון סריקה. הסיווג מתבצע באמצעות מכונת SVM (ראה 7.5) המופעלת על מתארי ה-HOG המחושבים בחלון הסריקה.

10.2 תיאור השיטה

10.2.1 חישוב מתאר מבוסס HOG עבור כל פיקסל

בשלב הראשון, מחשבים את תמונת הגרדיאנט של התמונה (ראה 10.1.1). לאחר חישוב זה, לכל פיקסל מותאם וקטור הגרדיאנט שלו, שהוא וקטור בעל שני מימדים ומורכב מהגודל שלו (המגניטודה) ומכיוון (°180-°0 אם הכיוון יכול להיות מיוצג גם עם סימן שלילי, °360-°0 אחרת). טווח המעלות בו משתמשים מחולק למקטעים המהווים את המחלקות של ההיסטוגרמות (לדוגמא [°360..°325],...,[°72..°37],[°36..°0]). לכל פיקסל בתמונה, מחשבים היסטוגרמת גרדיאנטים באופן הבא: מסתכלים על התא שבמרכזו נמצא הפיקסל (כזכור, תא הוא אזור פיקסלים ריבועי בגודל נתון). כל פיקסל בתא זה תורם ערך למחלקה אליו כיוון הגרדיאנט שלו שייך. הערך נקבע לפי המגניטודה של הגרדיאנט. כלומר עבור כל מחלקה בהיסטוגרמה סופרים כמה גרדיאנטים בתא נמצאים בטווח שלה, לפי משקלים שהם המגניטודות של הגרדיאנטים.

באופן כזה נוצרת היסטוגרמת גרדיאנטים עבור כל פיקסל בתמונה. בהינתן מספר המחלקות של ההיסטוגרמה, m, הוקטור המייצג את ההיסוגרמה הזאת הוא וקטור בעל m מימדים והוא הוקטור המתאר של הפיקסל.

10.2.2 חישוב מתאר ה-POEM

כפי שהוזכר בתחילת פרק זה, מאפייני POEM מורכבים משילוב של שתי שיטות ידועות – מתארי LBP ומתארי HOG. קיימים 2 הבדלים חשובים בשימוש במתארי HOG כאן מבשיטה שתיארנו בסעיף 10.1.2, המשתמשת במתארי HOG לזיהוי אובייקטים בתמונה:

1. בשיטה המקורית המשתמשת במתארי HOG, לכל תא קיים ייצוג של וקטור המהווה מתאר ייחודי של התא. לעומת זאת, בתהליך חישוב מתארי ה-POEM, לכל פיקסל בתמונה קיים וקטור ייחודי מבוסס HOG המתאר אותו. מתאר זה הוא הוקטור המתאים להיסטוגרמה של התא המכיל את אותו הפיקסל במרכזו.
2. השימוש בבלוקים הוא שונה – בלוקים משמשים לחישוב מתארי ה-LBP במקום לביצוע נורמליזציה.

מתאר POEM מחושב עבור כל פיקסל p והוא וקטור בעל m מימדים (כאשר m הוא מספר המחלקות בכל היסטוגרמה). הערך ה-i ב-m -יה הסדורה המייצגת את הוקטור $(1 \leq i \leq m)$, מתאר חישוב של מתאר ה-LBP על הפיקסל p כאשר הערכים בהם הוא משתמש הם הערכים המתאימים למחלקה שמספרה i בהיסטוגרמה המתאימה לפיקסל p, כלומר לערך המימד ה- i בוקטור ה-HOG המתאר את הפיקסל p. נתאר חישוב זה באופן פורמלי - בשלב הראשון מחשבים לכל פיקסל נתון, p, ולכל מחלקה, θ_i את הערך הבא:

$$POEM_{L,w,n}^{\theta_i}(p) = \sum_{j=1}^{n} f\left(S\left(I_p^{\theta_i}, I_{c_j(p)}^{\theta_i}\right)\right) 2^j \qquad (31)$$

כאשר:

- L הוא גודל הבלוק (בלוק יכול להיות ריבועי או עגול). בניסויים שנערכו בשיטה זו משתמשים בבלוק עגול משום שהוא מייצג באופן טוב יותר יחסי שכנות (פיקסלים במרחק של רדיוס מסוים הם הפיקסלים שדוגמים) כאשר L הוא הקוטר. הפיקסלים אותם דוגמים הם הפיקסלים שעל המעגל כלומר הפיקסלים במרחק $L/2$ מהפיקסל המרכזי.
- w הוא גודל התא (תא הוא ריבועי).
- n הוא מספר הפיקסלים המרכיבים את הבלוק.
- $c_j(p)$ הוא הפיקסל ה-j במספרו המקיף את את הפיקסל p.
- בכדי למצוא את $I_p^{\theta_i}$, מסתכלים על היסטוגרמת הגרדיאנטים שחושבה עבור p. $I_p^{\theta_i}$ הוא הערך של מחלקה θ_i בהיסטוגרמה זאת.
- $S(x, y)$ היא פונקציית ההבדל בין שתי מגניטודות x ו-y , במקרה זה פונקציית החיסור.
- $f(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases}$[1]

הערך $POEM_{L,w,n}^{\theta_i}(p)$ הוא בעצם חישוב מתאר LBP על הפיקסל p עבור המחלקה θ_i. עוברים על כל הפיקסלים שבסביבת הפיקסל p (הסביבה המעגלית שהיא הבלוק) באמצעות האינדקס j, וכל פיקסל שכן $c_j(p)$, מקבל 0 או 1 לפי ערך הסף של הפיקסל p, כאשר ערכי הפיקסלים $I_p^{\theta_i}, I_{c_j(p)}^{\theta_i}$ הם הערכים במחלקה θ_i של ההיסטוגרמה המתאימה להם. ערך זה הוא הפלט של הפונקציה $f\left(S\left(I_p^{\theta_i}, I_{c_j(p)}^{\theta_i}\right)\right)$ המחשבת אותו ע"י השוואה ל-0 של המרחק של

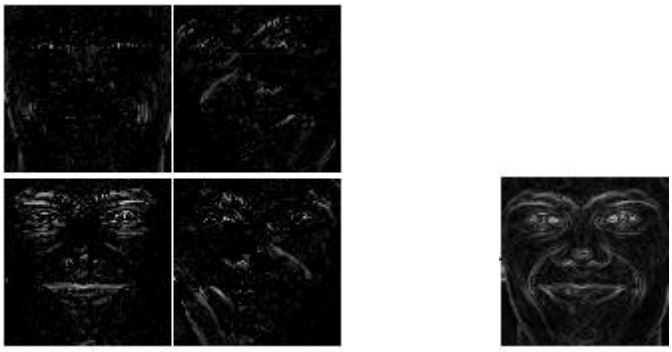
[1] למעשה, לא משתמשים ב-0 עבור ערך סף אלא בערך τ שהוא ערך מעט גדול מאפס. בשביל להגדיל את טווח הערכים שייחשבו זהים.

$I_p^{\theta_i}$ מ-$I_{c_j(p)}^{\theta_i}$. סכום הכפולות של ערכים אלה בחזקות עולות של 2 משרשר את ה0-ים וה-1-ים למספר הבינארי הייחודי עבור הפיקסל p והמחלקה θ_i.

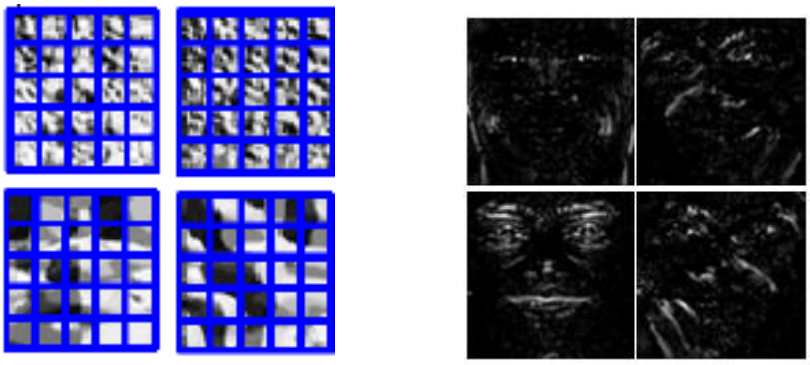
מתאר ה-POEM עבור פיקסל p הוא השרשור של כל המתארים הנ"ל בכל המחלקות השונות:

$$POEM_{L,w,n}(p) = \{POEM^{\theta_1}, \dots, POEM^{\theta_m}\} \qquad (32)$$

ניתן לחשב את אותם מתארים באופן הבא המודגם באיור 18 (א-ג): לאחר חישוב תמונת הגדריאנט הנקראת EMI (Edge Magnitude Image) לכל פיקסל מחושבות m תמונות שנסמנן ב-f_i , $1 \le i \le m$ כאשר בתמונה ה-i מופיעות רק מגניטודות של הכיוונים השייכים למחלקה מספר i (תמונות אלה נקראות גם הן EMIs). כעת, קל לחשב את $POEM_{L,w,n}^{\theta_i}(p)$ עבור כל מחלקה θ_i וכל פיקסל p. בכל תמונה שחושבה מקודם, f_i , כל פיקסל מכיל 0 או את ערך המגניטודה של גרדיאנט ששייך למחלקה ה-i . נחשב כעת m תמונות חדשות, g_i , $1 \le i \le m$ כך שכל פיקסל בתמונה מקבל את סכום ערכי הפיקסלים שבתא המכיל את אותו פיקסל במרכזו. תמונות אלה נקראות AEMIs (accumulated EMIs). נשים לב שכל פיקסל בתמונה כלשהי g_i מכיל בעצם את $I_p^{\theta_i}$. כעת כל שנותר לעשות הוא להפעיל את אופרטור ה-LBP על התמונות g_i ולקבל את הערכים $POEM^{\theta_1}, \dots, POEM^{\theta_m}$ עבור כל פיקסל, המרכיבים את מתאר ה-$POEM$ הסופי.



(ב) (א)



(ד) (ג)

איור 18 – תהליך חישוב ה-POEM

(א) תמונת הגדריאנט (EMI) (ב) 4 דוגמאות מתוך m התמונות f_i (EMIs), בכל תמונה מגניטודות רק של גרדיאנטים השייכים למחלקה i. (ג) 4 דוגמאות מתוך m התמונות g_i (AEMIs), תמונות בהן כל פיקסל מכיל את הערך של המחלקה ה-i , כלומר את סכום ערכי שכניו לתא בתמונות הEMI. (ד) תמונות ה-POEM המתקבלות ע"י חישוב מאפייני ה-LBP עבור כל אחת מתמונות ה-AEMI. (ערכי הפיקסלים בתמונות המתקבלות הם ערכי האפור המתאימים למספרים הבינארים המחושבים). מחלקים את תמונות ה-לאזורים שווים בגודלם, זרים ואשר מכסים את התמונות ולכל אזור כזה מחשבים היסטוגרמה.

10.2.3 זיווג תמונות פנים באמצעות מתארי POEM

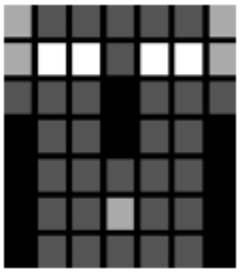
בסיום תהליך חישוב מתארי ה-POEM, מקבלים m תמונות הנקראות תמונות ה-POEM (איור 18 ד) מחלקים כל תמונה לאזורים ריבועיים שווים שאינם חופפים ואיחודם מהווה את התמונה השלמה, והיסטוגרמה מחושבת עבור על אזור כזה. כמו כן מרכיבים היסטוגרמה מכל ההיסטוגרמות של האזורים השונים הנקראת POEM-HS.

החלוקה לאזורים מאפשרת התייחסות שונה לאזורים יותר חשובים של הפנים כמו עיניים. למעשה, כל אזור מקבל משקל לפי מידת חשיבותו (ראה איור 19).

שתי תמונות פנים המתקבלות כקלט, מושוות באמצעות מידת כי בריבוע (chi square), כאשר w הוא וקטור המשקלים של האזורים השונים בתמונה:

$$\chi_w^2(S,M) = \sum_{i,j} w_j \frac{(S_{i,j}-M_{i,j})^2}{S_{i,j}+M_{i,j}} \qquad (33)$$

M_i ו-S_i הן ההיסטוגרמות המייצגות את שתי התמונות המושוות: $S_{i,j}$ הוא הערך במחלקה i בהיסטוגרמה של האזור שמספרו j. אם מידת הקרבה ביניהם מספיק גדולה, הפלט שמוחזר הוא חיובי.



(ב) **(א)**

איור 19 – משקלות האזורים

(א) תמונת פנים מחולקת ל-7X7 אזורים (ב) המשקל המוענק לכל אזור. הריבועים השחורים שבדרכ מהווים את הרקע מקבלים משקל 0. האפורים הכהים שהם בערך אזורי הפנים נטולי איברים הם בעלי משקל 1, האפורים הבהירים שהם אזורי השיער מקבלים משקל 2 והאזורים הלבנים המתאימים לעיניים מקבלים משקל 4.

11 תוצאות

בפרק זה ננתח את הישגיהן של השיטות בהן עסקנו בפרקים 8-10, ונערוך השוואה בין התוצאות שאליהן הגיעו. כמו-כן נשווה את התוצאות לתוצאותיו של ניסוי המודד את הישגיהם של בני אדם במשימת זיווג הפנים על סביבת LFW.

11.1 סביבות הניסויים

11.1.1 מאפיינים של דימויים ותיאורים

בניסויים של שתי השיטות שתוארו בפרק 8, המתבססות על מאפיינים של דימויים ותיאורים, מריצים תחילה תהליך עיבוד ראשוני על התמונות שהופך חלק מהתמונות כך שכל הפנים הם פרונטליים או פונים שמאלה. השימוש ב מאגר הנתונים LFW הוא בגישה המוגבלת. ישנו שימוש בשישה מאפיינים נוספים של תכונות העוסקים בזווית הפנים ובאיכות התמונה.

11.1.2 LDML ו-LDML+MkNN

נתייחס לתוצאותיהם של שני ניסויים [62] שמשתמשים שניהם בתמונות בגרסא הממוקדת של LFW (ראה 6.2):

השיטה הראשונה, (שנתייחס אליה בשם LDML), מציגה תוצאות בגישה המוגבלת. שיטה זו משלבת 8 מתארים שמשתמשים במרחקים שחושבו ע״י LDML. מתארים אלה הם SIFT [44], LBP [33], TPLBP ו-FPLBP [36], כאשר כל אחד נמצא בשימוש פעמיים – פעם אחת משתמשים בהיסטוגרמות המקוריות ובפעם השניה משתמשים בשורש של הנתונים בהיסטוגרמות. כזכור, מאפיינים אלה מייצרים היסטוגרמות אשר מתארות את תמונות הפנים ועל מנת לקבוע זיווג מודדים מרחק בין ההיסטוגרמות. מרחקים אלה, אם-כן מחושבים באמצעות LDML. אם-כן, המסווג הסופי מורכב מצירוף לינארי של 8 מתארים, כשהמקדמים של המתארים נקבעו ע״י תהליך למידה בשיטת הבחנה לוגיסטית לינארית (ראה 14.2.6).

על-מנת לבדוק את MkNN, נדרשת קבוצת למידה של איברים מתויגים. לכן מריצים את השיטה השניה, LDML+MkNN, בשיטה הבלתי מוגבלת בה כל תמונה מתויגת באמצעות שם. משתמשים באותו קלט של השיטה הקודמת (4 מתארים, פעם אחת עם ערכי ההיסטוגרמות המקוריים, פעם אחת עם השורש הריבועי שלהם) עם מטריצת המרחקים של LDML, LMNN ופעם אחת באמצעות השיטה MkNN כאשר מטריצת המרחקים שבה היא משתמשת היא LMNN (נמצא כי MkNN משיגה תוצאות הכי טובות כמטריצת המרחקים בה היא משתמשת היא מטריצת LMNN, כנראה משום שזוהי מטריצה המכוונת מראש לשפר את שיטת ה-kNN). כלומר הצירוף הלינארי כאן הוא צירוף של 24 ערכים, וכמו מקודם המקדמים נלמדים בשיטה ההבחנה הלוגיסטית.

11.1.3 POEM

התוצאות מוצגות על LFW בגישה המוגבלת. התמונות מיוצבות באמצעות השיטה LFW-a שתוארה בסעיף 6.3. בתהליך הבחינה נבדקה גם גרסא נוספת, אשר בה עבור כל זוג תמונות, הופכים אחת מהן סביב הציר האנכי המרכזי שלה (כלומר משתמשים בתמונת מראה שלה) ומשתמשים במרחק הנמוך ביותר מבין שני מרחקי ההיסטוגרמות שנוצרים. שיטה זו עוזרת להתמודד עם הבעות וזוויות פנים שונות ומשיגה אחוז מעט גבוה יותר של הצלחה (ראה POEM-Flip בטבלה 2).

11.2 השוואת מתאר ה-POEM עם מתארים ידועים אחרים

השוואה בין השיטות הקודמות שהראינו לשיטה המבוססת על POEM אינה הוגנת משום ש-POEM הוא מתאר ולכן מעניין להשוות אותו למתארים אחרים כגון SIFT וLBP. בשיטות של מאפייני דימויים ותיאורים ובשיטות המטריציוניות בהן דנו, מריצים מסווגים המבוססים על מתארים שונים במגוון של הרכבים וצירופים לינאריים.

בטבלה 2 ניתן לראות השוואה בין שימוש במתארים POEM ובין מתארים נוספים שימושיים. מדידת מרחקים בין היסטוגרמות של POEM ו- POEM-flip נעשיית באמצעות מידת כי בריבוע (chi square) ובשאר המתארים באמצעות מרחקים אוקלידיים. ניתן לראות ש-POEM משיג יתרון משמעותי על גבי כל שאר המתארים וגם על גבי שילוב של כולם יחדיו. אחוזי ההצלחה הם האחוזים הכי גבוהים מתוך כל הניסויים שנערכו עם ערכי סף שונים ונתונה גם סטיית התקן של הניסוי הכי מוצלח.

11.3 השוואת תוצאות הריצה והישגים של בני אדם בזיהוי

בפרק זה נשווה בין תוצאות ריצה של שיטות הדימויים והתיאורים, שיטות מטריציוניות, והישגיהם של בני אדם בזיווג פנים. קומר, ברג ועמיתיהם [57], ביצעו ניסויים בזיווג פנים על LFW ע״י בני אדם בכמה סביבות תנאים שונות. 10 אנשים שונים ענו על שאלונים לזיווג פנים והתוצאות הן ממוצעים של 10 ניסויים אלה. על מנת לייצר גרף מסוג ROC (ראה איור 20), התבקשו המשתתפים בניסויים לקבוע את הסיכוי של זוג פנים להיות פנים של אותו אדם, במקום להכריע באופן בינארי. הפנים עברו חיתוך כך שרק אזור הפנים נראה בתמונה גרסה זו של הניסוי היא הגרסה הנקראת Human, cropped באיור 20. מעניין לציין שכאשר הראו למשתתפים את התמונות המקוריות, התוצאות היו הרבה יותר מדויקות בגלל השימוש בהקשר של הרקע ושל השיער בזיהוי התמונה. למעשה, בניסוי שנערך שבו התמונות שהשתתפו היו תמונות של רקע בלבד עדיין אחוזי ההצלחה גבוהים יותר משל השיטות האוטומטיות שהצגנו (Human, inversed mask). ניסוי נוסף שנערך הוא על תמונות שעברו מיקוד (ראה 6.2). גרסא זו נקראת Human, funneled, ניסוי זה השיג את התוצאות הגבוהות ביותר.

מתאר	אחוזי הצלחה וסטיית תקן ממוצעים
LBP	68.24±67.90
Gabor	68.49±68.41
TPLBP	69.26±68.97
FPLBP	68.18±67.46
SIFT	69.12±69.86
LBP+Gabor+TPLBP+FPLBP+SIFT	75.21±0.55
POEM	74.00±0.62
POEM Flip	75.42±0.71

טבלה 2 – השוואה בין POEM ובין מתארים שונים בגישה המוגבלת

אחוזי הצלחה ממוצעים של POEM ושל מתארים שונים שימושיים וסטיות תקן ממוצעות (ראה 6.4) אחוזי ההצלחה אלה הם האחוזים הכי גבוהים שאליהם הצליחו להגיע ע"י הרצות עם ערכי סף שונים.

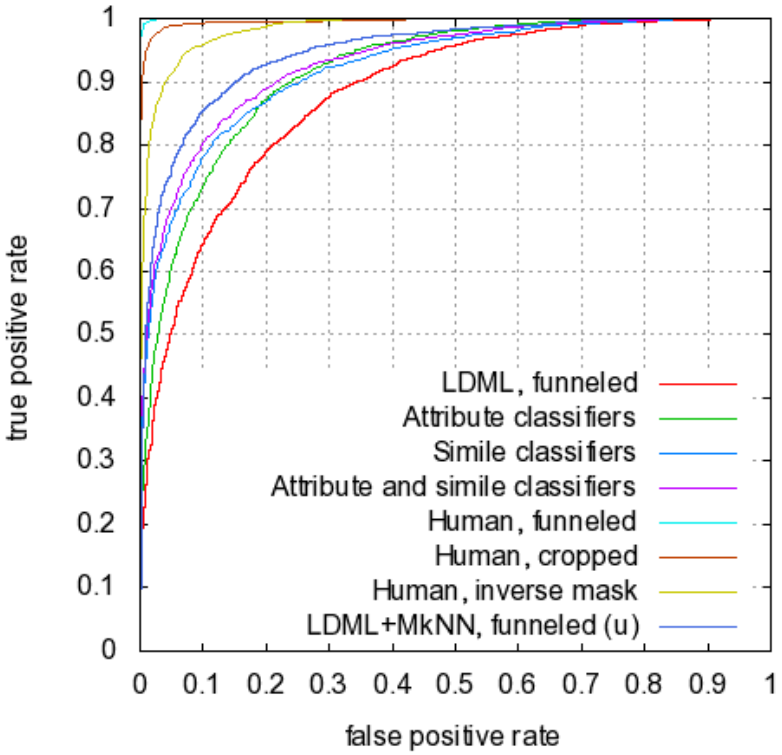
ניתן לראות בטבלה 3 את תוצאות השיטות השונות, בדומה להשוואה שערכנו בסעיף הקודם, אחוזי ההצלחה הם האחוזים הכי גבוהים מתוך כל הניסויים שנערכו עם ערכי סף שונים ונתונה גם סטיית התקן של הניסוי הכי מוצלח.

שיטה	סביבה	אחוזי הצלחה וסטיית תקן ממוצעים
LDML	מוגבלת	79.27±0.6
LDML+MkNN	בלתי מוגבלת	87.50±0.4
מאפיינים של תיאורים	מוגבלת	83.62±1.58
מאפיינים של דימויים	מוגבלת	84.14±1.31
מאפיינים של תיאורים ושל דימויים	מוגבלת	85.29±1.23

טבלה 3 – השוואת אחוזי הצלחה

באיור 20 ניתן לראות גרף ROC בו מופיעות השיטות שסקרנו ותוצאותיהם של בני אדם. גרף זה נוצר מתוצאות של ניסויים שונים עם ערכי סף שונים לסיווג. (למשל אם קובעים עבור זוג פנים

שהוא חיובי אם המרחק בין שני הפנים קטן מערך מסוים, ניתן לבדוק את אחוזי ההצלחה של הניסוי על טווח של ערכים כאלה). יש לציין שכל השיטות הורצו בשיטה המוגבלת, מלבד MkNN+LDML. ההשוואה היא רק בין השיטות שתארנו ולא מוצגות תוצאות של שיטות מוקדמות יותר או מאוחרות יותר, בכדי למנוע סרבול. תוצאות של כל השיטות מוצגות ב-[66]. מבין השיטות שפרסמו תוצאות על LFW בגישה הבלתי מוגבלת, MkNN+LDML משיגה תוצאות הכי טובות אחרי השיטה [67], שיטה מאוחרת יותר. מבין השיטות שנבדקו בגישה המוגבלת, מאפיינים של תיאורים ודימויים משיגה תוצאות הכי טובות אחרי השיטות [46],[68], גם הן שיטות מאוחרות יותר.



איור 20 – גרף ROC

השוואה בין תוצאות של בני אדם, מאפיינים של תיאורים ודימויים ושיטות למידה. הגרף נוצר ע"י חישוב אחוזי הצלחה לבין ערכי סף משתנים. כל השיטות המושוות בגרף זה הורצו בשיטה המוגבלת מלבד LDML+MkNN.

12 אפליקציה לזיהוי פנים בוידיאו

כחלק מעבודה זו פותחה אפליקציה המדגימה זיהוי פנים מוידיאו בזמן אמת. עבודה זו הוצגה ביום מחקר של האוניברסיטה הפתוחה, מאי 2010 [69].

האפליקציה מצלמת וידיאו ומציגה אותו על המסך בזמן אמת. בכל מספר קבוע של פריימים (מספר זה נקבע בשורת ההרצה של התוכנית) מתבצע ניסיון לאתר פנים. בכל פעם שאותרו פנים, מופעל תהליכון (process) שמנסה לזהות אותם ברקע. המצלמה ממשיכה לצלם ולשדר את הוידיאו על המסך, במידה ומתקבל זיהוי מודפס על המסך משפט אקראי כלשהו המכיל את השם שזוהה.

הזיהוי מתבצע באמצעות ממשק חדשני של Face.com - חברה ישראלית המספקת ממשק לזיהוי פנים בתמונה, באמצעות חשבון בפייסבוק ומאגר הפנים של החברים של המשתמש.

איתור הפנים ותפעול הוידיאו מתבצעים באמצעות ספריות הקוד הפתוח OpenCV [48]. ספריות אלה ממשות איתור פנים באמצעות האלגוריתם של Viola and Jones [2].

12.1 הממשק של Face.com

Face.com [70] היא חברת סטארט-אפ ישראלית שהוקמה ב-2007 ומספקת אפליקציות לתיוג אוטומטי של תמונות בפייסבוק (אשר אינן מוכרות למשתמש) ומציאת תמונות של ידועניים בטוויטר. לאחרונה הודיעה החברה על פתיחת שירותיה למפתחים חיצוניים. Face.com מאפשרת להשתמש באלגוריתמים שלה לזיהוי פנים באמצעות למידה מתוך תיוג בפייסבוק, בטוויטר, או באמצעות למידה מתוך מאגר תמונות פרטי. האפליקציה המוצגת כאן משתמשת בפייסבוק כפלטפורמה לזיהוי הפנים.

זיהוי פנים באמצעות הפייסבוק נעשה רק מתוך רשימת חברים של משתמש שנתן הרשאה להשתמש ברשימה זו. לכן יש ליצור אפליקצית פייסבוק אשר בזמן התקנתה מבקשת הרשאה מהמשתמש לגשת לפרטי המשתמש שלו (במקרה זו תמונות של חברים). בכדי להשתמש בממשק של Face.com, נדרש לבצע רישום חד פעמי של האפליקציה באתר של Face.com וברישום זה מקשרים את האפליקציה לאפליקציית הפייסבוק. אם ההרשאה ניתנה ע״י משתמש הפייסבוק, ניתן להפעיל את הפונקציות של Face.com עבור זיהוי פנים, כשמאגר הפנים המשמש ללמידה הוא מאגר הפנים של החברים של המשתמש (המורכב מסך כל התמונות המתוייגות של החברים). והזיהוי יהיה מבין חברים אלה.

במהלך תהליך הרישום הראשוני מקבל מפתח האפליקציה שני מפתחות. לכל פונקציה בממשק של Face.com יש כתובת אינטרנט ייחודית והשימוש בממשק של Face.com נעשה באמצעות העלאה (upload) של מחרוזת הכוללת מפתחות אלה, פרמטרים לפונקציה, ותמונה, לכתובת הספציפית של הפונקציה בה רוצים להשתמש.

הזיהוי מתקבל גם הוא דרך האינטרנט והוא למעשה מכיל מחרוזת, כתובה בפורמט JSON[2] המכילה רשימה של פרטים לגבי זיהוי הדמויות שבתמונה. בין השאר היא מכילה רשימה של תגים (tags), תג עבור כל דמות שזוהתה בתמונה (ניתן לזהות יותר מאדם אחד בכל תמונה). כל תג מכיל רשימה של כמה מספרי זיהוי של אנשים. לכל מספר זיהוי ציון המבטא את רמת הביטחון (confidence) של הזיהוי. בנוסף עבור כל דמות קיים מספר סף (threshold). ערכי ביטחון שמעל ערך זה נחשבים כערכים של זיהויים סבירים. הזיהוי בעל ערך הביטחון הגבוה ביותר הוא הזיהוי הסופי שנבחר.

ניתן להשתמש בממשק של Face.com גם באופן אינטראקטיבי באמצעות אתר האינטרנט [71]. לדוגמא, באיור 21. מוצגת תמונה לאחר הפעלת הפונקציה של זיהוי פנים של Face.com.

(א)

[2] JSON - JavaScript Object Notation הוא פורמט קריא המשמש להעברת מאגרי מידע פשוטים, בעיקר ברשת.

Attributes:	
face:	true (2.81202)
gender:	female (95%)
glasses:	false (60%)
smiling:	true (98%)
Rotations:	
roll:	-22.02°
yaw:	-10.99°
pitch:	-1.76°
High Confidence Recognitions:	
Tal Agmon:	96
Low Confidence Recognitions:	
Melisa Sametband:	8
Lilach Zilkha:	7

(ב)

(ג)

איור 21 – זיהוי פנים בface.com

(א) תמונה לאחר הפעלה של הפונקציה face.recognize באתר האינטרנט של Face.com [71]. (ב) מידע על הזיהוי של הדמות Tal Agmon. (ג) חלק מהמחרוזת Jason שהתקבלה המתאר את תג הזיהוי של Tal Agmon. בזיהוי זה הדמות שזוהתה כ-Tal Agmon זוהתה בביטחון של 96, והיא זוהתה כMellisa Samentband בביטחון של 8 וכ-Lilach Zilka בביטחון של 7. ערך הסף הקובע איזה זיהוי נחשב זיהוי סביר הוא 52 כך שרק הזיהוי של Tal Agmon הוא סביר.

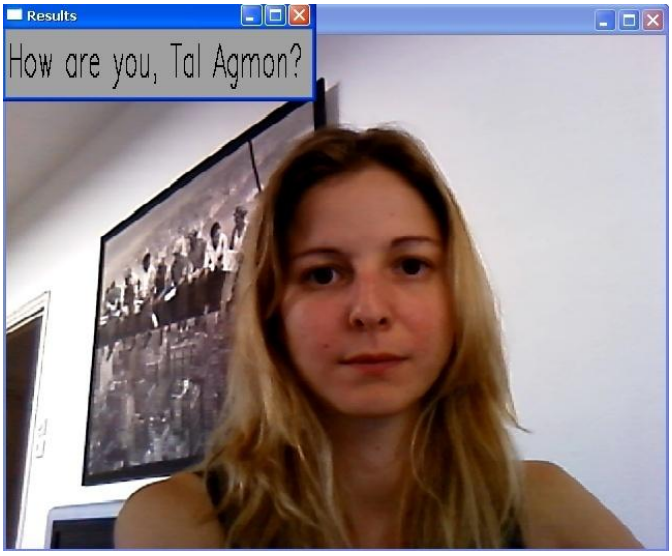
12.2 שיפור יכולת הזיהוי

על-מנת לשפר את יכולת הזיהוי, ניתן להריץ את האפליקציה עם ערכי קלט הקובעים כל כמה פריימים (frames) יפעיל האלגוריתם מנגנון לאיתור פנים וכמה פעמים תופעל הפונקציה לזיהוי פנים של Face.com על כל פריים כנ"ל (הזיהוי הראשון שמצליח לעבור את ערך הסף נחשב). ביום המחקר באוניברסיטה הפתוחה, הורצה האפליקציה כאשר כל 10 פריימים הופעל מנגנון לאיתור

פנים והפונקציה לזיהוי פנים מופעלת פעם אחת על כל פריים. מאחר ופונקציה זו מסתמכת על מהירות ההעלאה וההורדה של חיבור האינטרנט, הזיהוי התבצע באופן מעט איטי ולכן הפונקציה הופעלה פעם יחידה.

12.3 הרצת האפליקציה ביום המחקר באו"פ

בהרצת האפליקציה ביום המחקר באוניברסיטה הפתוחה, זוהה נכונה מספר גבוה של אנשים. עם זאת היו מספר אנשים שאותם המערכת לא זיהתה כללה או לא זיהתה נכונה. עבור הרבה מן האנשים שלא זוהו נכונה לא מופיעות הרבה תמונות בחשבון הפייסבוק שלהם, מה שמוביל להנחה שהאלגוריתם לזיהוי פנים של face.com עדיין התקשה במקרים מסוימים לזהות פנים מדוגמא יחידה או ממספר קטן של דוגמאות. יש לציין שלרוב התמונות המופיעות בפייסבוק שונות מאוד מהתמונות שצולמו בוידיאו באפליקציה משום שהתאורה שונה, המצלמה שונה, ואף ההבעות הן לרוב שונות. ניתן לראות הדפסה של המסך כשהאפליקציה רצה, מיד לאחר זיהוי באיור 22.



איור 22 – אפליקציה לזיהוי פנים בוידאו

הדפסת מסך הפלט של האפליקציה, מיד לאחר זיהוי.

13 מסקנות וכיווני מחקר עתידיים

בעבודה זאת דנו בשלוש שיטות חדישות לזיווג פנים בסביבה חסרת אילוצים. שלושת השיטות משיגות שיפור משמעותי באחוזי ההצלחה והישגים מבין הגבוהים ביותר בפרוייקט ה-LFW. הקשיים העיקריים שעליהם יש להתגבר בכדי להשיג אחוזי הצלחה גבוהים הם שינויים בגיל, בתאורה, בהבעות, בזוויות הפנים ובמאפייני פנים נוספים כמו זקן ומשקפיים.

השיטות בהן דנו מתמודדות עם קשיים סביבתיים אלה באמצעים שונים: השימוש במאפיינים של תיאורים ודימויים, כלומר שימוש בתכונות ויזואליות ובדימיון או חוסר דימיון לאדם אחר כבסיס מאפשר ללכוד מאפיינים יחודיים של פנים (למשל גוון עור או צורה אליפסית של פנים) ומעלה את העמידות בפני שינויי הבעה וזווית הפנים. השיטות המטריציוניות שהצגנו מתמודדות עם האי-לינאריות של המרחב הנפרש ע״י פנים ע״י למידת המרחקים המאפיינים בצורה יותר מדויקת את צורת המרחב במקום שימוש במרחקים אוקלידיים. השימוש שהצגנו במאפייני POEM עמיד יחסית בפני שינויי תאורה משום שמאפיינים אלה מתבססים על גרדיאנטים ולא על ערכי הפיקסלים. כמו-כן השימוש ביחסים בין פיקסלים בכמה רזולוציות שונות מאפשר עמידות גבוהה יותר בפני שינויי הבעה וזווית.

עם זאת, ראינו שאחוזי ההצלחה עדיין אינם גבוהים כמו אחוזי ההצלחה של בני אדם במשימה זאת, ועל כן האתגר בתחום זה הוא עדיין גדול. לשם כך יש למצוא שיטות חדשות לזיהוי פנים, לשלב שיטות ישנות או לשכלל שיטות קיימות, על מנת להשיג עמידות בפני מגוון דרכי ההצגה השונים של פנים אשר תעלה את אחוזי ההצלחה בניסויים.

עבור מתארי POEM כיוון מחקר עתידי מעניין הוא לבחון את השימוש בהם בשיטות המשתמשות במתארים של מאפיינים מקומיים שונים כמו השיטות שתיארנו בעבודה זו המשתמשות בSIFT, LBP ובואריציות של LBP. באופן כללי, כיווני מחקר עתידיים הם: המשך שיפורם של אחוזי ההצלחה ע״י מחקר של שיטות למידה חישובית היכולות להתאים לבעיה זו ומציאת שילובים חדשים של שיטות ישנות שמייצרים תוצאות טובות יותר. חקירה של אלגוריתמים שפועלים לפני שלב הזיווג על מנת לנטרל השפעות של תאורה, רקע וזוויות פנים והמשך מחקר ואנליזה העוסקים בהשפעות של הבעות פנים ושינויים בפנים המשפיעים לרעה על זיווג פנים ועל דרכי התמודדות איתם.

14 נספח מתמטי

14.1 מושגים באלגברה לינארית

14.1.1 מטריצה חיובית לחלוטין

מטריצה ריבועית מסדר $n \times n$, ממשית וסימטרית M, היא מטריצה חיובית לחלוטין (positive definitive) אם $z^T M z > 0$ עבור כל וקטור $0 \neq z \in \mathbb{R}^n$.

14.1.2 מרחב לינארי ותת-מרחב לינארי

מרחב לינארי (linear space), או מרחב וקטורי (vector space) מעל שדה F, הוא קבוצה של איברים V אשר עבורה מתקיימות הדרישות הבאות:

1. על V מוגדרת פעולת החיבור שהיא פעולה בינארית המקיימת את התכונות:
   א. סגירות: לכל $u, v \in V$ מתקיים $u + v \in V$.
   ב. אסוציאטיביות: לכל $u, v, w \in V$ מתקיים $(u + v) + w = u + (v + w)$.
   ג. קומוטטיביות: לכל $u, v \in V$ מתקיים $u + v = v + u$.
   ד. קיום איבר ניטרלי: קיים $0 \in V$ כך שלכל $v \in V$ מתקיים $v + 0 = v$.
   ה. קיום איברים נגדיים: לכל $v \in V$ קיים $-v \in V$ כך שמתקיים $v + (-v) = 0$.
2. בין איברי V לאיברי השדה F מוגדרת פעולת הכפל בסקלר המקיימת את התכונות:
   א. סגירות: לכל $v \in V$ ו-$a \in F$ מתקיים $av \in V$.
   ב. דיסטריבוטיביות מעל החיבור ב-V: לכל $u, v \in V$ ו-$a \in F$ מתקיים
   $$a(u + v) = au + av.$$
   ג. דיסטריבוטיביות מעל החיבור ב-F: לכל $v \in V$ ו-$a, b \in F$ מתקיים
   $$(a + b)v = av + bv.$$
   ד. אסוציאטיביות: לכל $v \in V$ ו-$a, b \in F$ מתקיים $(ab)v = a(bv)$.
   ה. זהות: לכל $v \in V$ מתקיים $1v = v$ (1 הוא איבר היחידה של F).

בגלל השימוש הנרחב במושג מרחב לינארי, לעיתים מקצרים ומשתמשים רק במונח מרחב.

תת-קבוצה W , של מרחב לינארי V מעל שדה F נקראת תת-מרחב (subspace) של V אם W עצמה מהווה מרחב לינארי מעל F, ביחס לפעולות החיבור והכפל בסקלר של המרחב V.

14.1.3 טרנספורמציה (העתקה) לינארית

טרנספורמציה לינארית היא פונקציה ממרחב וקטורי אחד למרחב וקטורי אחר אשר משמרת את שתי התכונות:

1. חיבור: לכל $u, v \in V$ מתקיים $T(u+v) = T(u) + T(v)$
2. כפל: לכל $v \in V$ ו-$a \in F$ מתקיים $T(av) = aT(v)$

עובדה חשובה היא שאם A היא מטריצה מסדר $m \times n$, אז A מגדירה טרנספורמציה לינארית מ-$\mathbb{R}^n$ ל-$\mathbb{R}^m$ על ידי כפל מטריצות מימין. ניתן לייצג כל טרנספורמציה לינארית בין מרחבים ממימד סופי בדרך זו.

14.1.4 בסיס של מרחב לינארי

קבוצת וקטורים נקראת תלויה לינארית אם ניתן להציג את אחד מהוקטורים שלה כצירוף לינארי של וקטורים אחרים בקבוצה.

קבוצת וקטורים S נקראת קבוצה פורשת אם כל וקטור במרחב ניתן להצגה כצירוף לינארי של הוקטורים השייכים ל-S.

קבוצה פורשת מינימלית נקראת בסיס (base). כל שתי קבוצות פורשות של אותו מרחב הן בעלות אותו גודל וגודל זה הוא המימד של המרחב.

14.1.5 וקטור עצמי וערך עצמי

תהי A היא מטריצה ריבועית מסדר n מעל שדה F ו-$v \in F^n$ וקטור שונה מאפס. אם קיים סקלר $\lambda \in F$ כך ש-$Av = \lambda v$ אזי λ נקרא ערך עצמי (eigenvalue) ו-v נקרא וקטור עצמי (eigenvector) השייך לערך העצמי λ.

מכאן, שהפעלת מטריצה של וקטור עצמי לא משנה את הכיוון שלו, אלא רק את המגניטודה שלו. הפעלת מטריצה על וקטור כלשהו תטה אותו לכיוון הוקטור העצמי עם הערך העצמי הגדול ביותר.

14.2 מושגים בסטטיסטיקה

14.2.1 מטריצת שונות

שונות (variance) היא מדד לפיזור של משתנה מקרי סביב התוחלת שלו ומוגדרת כך:

$$var(X) = E((X - E(X))^2) \tag{34}$$

שונות משותפת (covariance) היא מדד לקשר בין שני משתנים מקריים. השונות המשותפת היא שלילית כאשר הם משתנים בכיוונים מנוגדים וחיובית כאשר הם נוטים להשתנות באותו כיוון. אם הם בלתי תלויים, השונות המשותפת היא אפס. ובאופן פורמלי:

$$cov(X,Y) = E[(X - E(X))(Y - E(Y))] = E(X,Y) - E(X)E(Y) \quad (35)$$

מטרימת שונות (covariance matrix) מוגדרת עבור וקטור של משתנים מקריים. בהינתן וקטור של משתנים מקריים $X = \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix}$, אזי מטריצת השונות היא המטריצה שהכניסה ה-(i,j) שלה מכילה את הערך $cov(X_i, X_j)$.

14.2.2 התפלגות נורמלית

התפלגות נורמלית היא ההתפלגות ששימושיה הם הנפוצים ביותר בסטטיסטיקה: התפלגויות רבות בטבע מתנהגות בקירוב כהתפלגות נורמלית, ובמקרים רבים משתמשים בהתפלגות נורמלית כקירוב כאשר ישנו מידע מוגבל על ההתפלגות. משפט הגבול המרכזי קובע כי לפונקציות רבות של מדגם גדול הנלקח מהתפלגות כלשהי יש התפלגות נורמלית. למשל, אם מדגם נלקח מהתפלגות כלשהי בעלת שונות סופית, הממוצע שלו מתפלג בקירוב נורמלית .

משתנה מקרי רציף בעל תוחלת μ וסטיית תקן σ הוא בעל התפלגות נורמלית אם פונקציית הצפיפות שלו היא מהצורה:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right) \quad (36)$$

ומסמנים $X \sim N(\mu, \sigma^2)$. כאשר $\mu = 0$ ו-$\sigma = 1$ אז ההתפלגות נקראת נורמלית סטנדרטית.

14.2.3 הערכה של סטיית תקן

במקרים בהם לא ניתן לדגום את כל אברי קבוצת הניסוי ולכן לא ניתן לתת סטיית תקן מדויקת, משתמשים בהערכה של סטיית התקן, ע"י דגימה של כמה איברים אקראיים מתוך הקבוצה. ישנן כמה שיטות מקובלות לחישוב סטיית תקן מדגמית, השכיחה ביניהן נקראת סטיית תקן מדגמית (sample standard deviation) המחושבת באופן הבא:

$$s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2} \quad (37)$$

כאשר $\bar{x}$ הוא הממוצע ו-$x_1, \ldots, x_n$ הם איברי המדגם.

14.2.4 אנטרופיה

אנטרופיה (Entropy) היא מידה של חוסר ודאות של מרחב מדגם ומוגדרת ע"י:

$$\sum_{i=1}^{n} p_i log\left(\frac{1}{p_i}\right) = -\sum_{i=1}^{n} p_i log(p_i) \quad (38)$$

(כאשר מרחב המדגם מכיל n מאורעות וההתסברות לקבלת מאורע i היא p_i). אנטרופיה מקסימלית מתקבלת כאשר $p_i = 1/n$ לכל i - מצב בו אי הודאות לגבי התרחשותם של המאורעות היא הכי גבוהה. אנטרופיה יחסית (relative entropy), הנקראת גם The Kullback Leibler Divergence, היא מדד למידת הדמיון הסטטיסטי בין התפלגויות ומוגדרת עבור שתי התפלגויות, P, Q ומשתנה מקרי X בדיד באופן הבא:

$$KL(p||q) = \sum_x p(x) log \frac{p(x)}{q(x)} \quad (39)$$

ועבור משתנה מקרי רציף:

$$KL(p||q) = \int p(x) log \frac{p(x)}{q(x)} dx \quad (40)$$

14.2.5 שיטת הריבועים הפחותים

בשיטת הריבועים הפחותים (minimal squares;) היא שיטת אומדן סטטיסית שבה מנסים למצוא את הפרמטרים שירכיבו פונקציית המודל הכי מתאימה לקבוצה של מדידות נתונות $(x_i, y_i), i = 1, \dots, n$. פונקציית המודל היא מהצורה $f(x, \beta)$ כאשר β הוא וקטור של m פרמטרים אותם צריך למצוא.לשם כך מחשבים את השאריות שהן המרחקים של המדדים האמיתיים מתוצאות פונקציית המודל:

$$r_i = y_i - f(x_i, \beta) \quad (41)$$

ומנסים להביא למינימום את סכום ריבועי השאריות:

$$S = \sum_{i=1s}^{n} {r_i}^2 \quad (42)$$

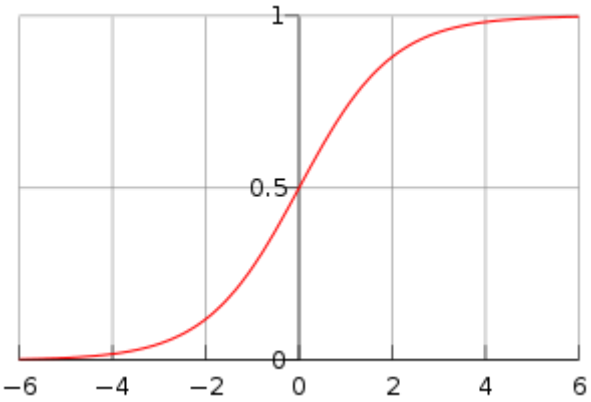
לשם כך יש להשוות את כל m הגרדיאנטים של סכום זה לאפס:

$$\frac{\partial S}{\partial \beta_j} = 2\sum_i r_i \frac{\partial r_i}{\partial \beta_j} = 0, j = 1, \dots, m \quad (43)$$

ניתן לפתור משוואות אלה בדרכים שונות, לפי סוג פונקציית המודל הספציפי לבעיה.

14.2.6 סיגמואיד ורגרסיה לוגיסטית

סיגמואיד, או פונקציה לוגיסטית (sigmoid, logistic function) היא פונקציה מהצורה $\frac{1}{1+e^{-x}}$. התחום הוא כל מספר ממשי והטווח הוא התחום שבין 0 ל-1. פונקציה לוגיסטית היא בעלת הצורה "S" (ראה איור 23) והן מאוד שימושיות בהרבה תחומים וביניהם שיטת הרגרסיה הלוגיסטית (logistic regression).



איור 23 – פונקציה לוגיסטית

בשיטת הרגרסיה לוגיסטית מנסים להעריך את ההסתברות להתרחשותו של מאורע. לשם כך נעזרים בכמה משתנים בלתי תלויים המסייעים לנבא הסתברות זו (למשל ניתן לנבא את ההסתברות שזוג פנים מייצג פנים של אותו אדם לפי הנתונים על גגון העור, צורת הפנים, המגדר וכו'). ההערכה מתבצעת באמצעות פונקציית סיגמואיד $f(z)$ כאשר z הוא ביטוי המכיל את משתני הקלט הבלתי תלויים ומוגדר כך:

$$z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k \qquad (44)$$

כל אחד מהמקדמים $\beta_1, \ldots, \beta_k$ מתארים את מידת ההשפעה של המשתנים $x_1, \ldots, x_k$ (מקדם חיובי משמעותו שהמשתנה מגדיל את ההסתברות ושלילי קובע כי הוא מקטין אותה).

14.2.7 אומדן נראות מקסימלית

נראות מקסימלית (Maximum likelihood ML) היא שיטה אשר בהינתן קבוצת למידה ומודל הסתברותי, מוצאת את ערכי הפרמטרים למודל שהופכים אותו להיות הכי מתאים עבור קבוצת למידה זו. באופן פורמלי, בהינתן קבוצת למידה, כלומר קבוצת מאורעות בלתי תלויה $x_1, \ldots, x_n$, מחפשים את הפרמטרים $\theta = \{\theta_1, \ldots, \theta_n\}$ שיביאו למקסימום את הנראות שנתונה על ידי:

$$L(x_1, \ldots, x_n|\theta) = \prod_{i=1}^{n} P(x_i|\theta) \qquad (45)$$

כלומר ההסתברות שיקרו כל המאורעות בקבוצת הלמידה. בהינתן הפרמטרים θ. אומדים את הפרמטרים באמצעות פתירת המשוואה:

$$\hat{\theta} = arg_{\theta}\, max\, [L(x_1, \dots, x_n|\theta)] \qquad (46)$$

הדרך השכיחה לפתרון היא באמצעות השוואה של הגרדיאנט של $L(x_1, \dots, x_n|\theta)$ לאפס.

15 ביבליוגרפיה

.1 Learned-Miller, G.B.H.a.M.R.a.T.B.a.E., Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments. Tech. Rep. 7-49, University of Massachusets Amherst, 2007. 07(49)

.2 Viola, P. and M. Jones, Rapid object detection using a boosted cascade of simple features. Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on, 2001. 1: p. I-511-I-518 vol.1.

.3 Kirby, L.S.a.M., Low-dimensional procedure for the characterization of human faces endnote. Journal of the Optical Society of America 1987. 4(3): p. 519–524

.4 Sirovich, M.K.a.L., Application of the Karhunen-Loeve procedure for the characterization of human faces. Pattern Analysis and Machine Intelligence ,IEEE Transactions on 1990. 12(1): p. 103–108.

.5 Pentland, M.T.a.A., Face recognition using eigenfaces. Proc. IEEE Conference on Computer Vision and Pattern Recognition, 1991: p. 586–591.

.6 Matthew Turk, A.P., Eigenfaces for recognition. J. Cognitive Neuroscience, 1991. 3(1): p. 71-86.

.7 Lawrence, S., Giles, C.L., Tsoi, A. and Back, A., Face recognition: A convolutional neural-network approach. IEEE Trans. on Neural Networks, 1997. 8(1): p. 98-113.

.8 Tan X., C.S.C., Zhou Z.-H., and Zhang F. , Recognizing partially occluded, expression variant faces from single training image per person with SOM and soft kNN ensemble. IEEE Transactions on Neural Networks, 2005. 16(4): p. 875-886.

.9 B., J.A.K.a.C., Dimensionality and Sample Size Considerations in Pattern Recognition Practice. Handbook of Statistics, 1987. 2: p. 835 - 855.

.10 A.K., R.S.J.a.J. and . Small sample size effects in statistical pattern recognition: recommendations for practitioners. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1991. 13(2): p. 252-264.

.11 Phillipls, P.J., Support vector machines applied to face fecognition. Adv. Neural Inform. Process. Syst., 1998. 11(3): p. 809.

.12 X. Tan, S.C., Z.-H. Zhou, and F. Zhang, Face recognition from a single image per person: a survey. Pattern Recognition, 2006. 39(9): p. 1725–1745.

.13 B. S. Manjunath, R.C., C. von der Malsburg, A feature based approach to face recognition. IEEE Computer Society Conference, 1992: p. 373-378.

.14 Jie, W., et al., On solving the face recognition problem with one training sample per subject. Pattern Recogn., 2006. 39(9): p. 1746-1762.

.15 G. B. Huang, M.R., T. Berg, and E. Learned-Miller, Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments. Tech. Rep ,49-7 .University of Massachusets Amherst, 2007.

.16 DARPA, http://www.darpa.mil/index.html.

.17 Lior Wolf, T.H., Yaniv Taigman, Effective Unconstrained Face Recognition by Combining Multiple Descriptors and Learned Background Statistics. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2010.

.18 Bolle, A.W.S.a.R.M., Face recognition and its applications, in Biometric Solutions for Authentication. 2002 E-World, D.Zhang ed. Kluwer Academic.

.19 Goldstein, A.J., L.D. Harmon, and A.B. Lesk, Identification of human faces. Proceedings of the IEEE, 1971. 59(5): p. 748-760.

.20 Jian Yang, D.Z., Alejandro F. Frangi, Jing-yu Yang Two-Dimensional PCA: A New Approach to Appearance-Based Face Representation and Recognition. 2004. 26(1): p. 131-137.

.21 Yang, J., et al., KPCA plus LDA: a complete kernel Fisher discriminant framework for feature extraction and recognition. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2005. 27(2): p. 230-244.

.22 Christopher, M.B., Bayesian PCA, in Proceedings of the 1998 conference on Advances in neural information processing systems II. 1999, MIT Press.

.23 Etemad, K. and R. Chellappa, Discriminant analysis for recognition of human face images. J. Opt. Soc. Am. A, 1997. 14(8): p. 1724-1733.

.24 Lu, J ,.K.N. Plataniotis, and A.N. Venetsanopoulos, Face recognition using LDA-based algorithms. IEEE Transactions on Neural Networks, 2003. 14(1): p. 195-200.

.25 Belhumeur, P., J. Hespanha, and D. Kriegman, Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997. 19(7): p. 711-720.

.26 Tenenbaum, J., V. Silva, and J. Langford, A Global Geometric Framework for Nonlinear Dimensionality Reduction. Science, 2000. 290(5500) : p. 2319-2323.

.27 Saul, S.T.R.a.L.K., Nonlinear dimensionality reduction by locally linear embedding. Science, 2000. 290: p. 2323-2326.

.28 Saul, S.T.R.a.L.K., An introduction to locally linear embedding. AT&T 2000.

.29 Belkin, M. and P. Niyogi. Using Manifold Structure for Partially Labelled Classification. in NIPS. 2002.

.30 Michail, V., et al., Non-linear dimensionality reduction techniques for classification and visualization, in Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. 2002, ACM: Edmonton, Alberta, Canada.

.31 Cuihong, Z. and Y. Gelan, Research of Face Recognition Based on Locally Linear Embedding, in Proceedings of the 2009 Second International Conference on Computer and Electrical Engineering - Volume 02. 2009, IEEE Computer Society.

.32 Nathan, M., B. Christian, and K.T. John, Face Recognition with Weighted Locally Linear Embedding, in Proceedings of the 2nd Canadian conference on Computer and Robot Vision. 2005, IEEE Computer Society.

.33 Ahonen, T., et al., Face description with local binary patterns: Application to face recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2006. 28: p. 2037-2041.

.34 Ahonen, T., A. Hadid, and M. Pietikainen, Face Recognition with Local Binary Patterns, in Computer Vision - ECCV 2004. 2004. p. 469-481.

.35 Marko, H., et al., Description of interest regions with local binary patterns. Pattern Recogn., 2009. 42(3): p. 425-436.

.36 Wolf, L., T. Hassner, and Y. Taigman. Y.: Descriptor based methods in the wild. in In: Faces in Real-Life Images Workshop in ECCV. (2008) (b) Similarity Scores based on Background Samples.

.37 I. Fogel, D.S., Gabor filters as texture discriminator. Biological Cybernetics, 1989. 61(2): p. 103-113.

.38 S.E. Grigorescu, N.P.a.P.K., Comparison of texture features based on Gabor filters. IEEE Trans. on Image Processing, 2002. 11(10): p. 1160-1167.

.39 The University of Edinburgh, School of Informatic, http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/TRAPP1/filter.html.

.40 Choi, W., et al., Simplified Gabor wavelets for human face recognition. Pattern Recognition, 2008. 41(3): p. 1186-1199.

.41 Jianke, Z., I.V. Mang, and M. Peng Un, Gabor Wavelets Transform and Extended Nearest Feature Space Classifier for Face Recognition, in Proceedings of the Third International Conference on Image and Graphics. 2004, IEEE Computer Society.

.42 Liu, C., Gabor-Based Kernel PCA with Fractional Power Polynomial Models for Face Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2004. 26(5): p. 572-581.

.43 Pinto, N., J.J. DiCarlo, and D.D. Cox, How far can you get with a modern face recognition test set using only simple features? Computer Vision and Pattern Recognition, IEEE Computer Society Conference on, 2009. 0: p. 2591-2598.

.44 Lowe, D.G., Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision, 2004. 60(2): p. 91 - 110.

.45 Lowe, D., Object Recognition from Local Scale-Invariant Features. Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on, 1999. 2: p. 1150-1157 vol.2.

.46 Wolf, L., T. Hassner, and Y. Taigman. Similarity Scores based on Background Samples. in Asian Conference on Computer Vision (ACCV). 2009.

.47 Tamara L. Berg, A.C.B., Jaety Edwards, Michael Maire, Ryan White, Yee-Whye Teh, Erik Learned-Miller and D.A. Forsyth, Names and faces in the news, in In proc. CVPR. 2004, IEEE Computer Society. p. 848-854.

.48 OpenCV, http://opencv.willowgarage.com/wiki.

.49 Peng, W., T. Lam Cam, and J. Qiang, Improving Face Recognition by Online Image Alignment, in Proceedings of the 18th International Conference on Pattern Recognition - Volume 01. 2006, IEEE Computer Society.

.50 Huang, G.B.J., V.; Learned-Miller, E.;, Unsupervised Joint Alignment of Complex Images, in Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference 2007 Rio de Janeiro p. 1 - 8

.51 Learned-Miller., E., Data driven image models through continuous joint alignment. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 2006. 28(2): p. 236-250.

.52 Yoav, F. and E.S. Robert, A decision-theoretic generalization of on-line learning and an application to boosting, in Proceedings of the Second European Conference on Computational Learning Theory. 1995, Springer-Verlag.

.53 Wu, J., J. Rehg, and M. Mullin. Learning a Rare Event Detection Cascade by Direct Feature Selection. in In NIPS. 2003.

.54 VAPNIK, V., Statistical Learning Theory. 1998, New York: Wiley.

.55 Vapnyarskii, I.B., Lagrange multipliers, in Encyclopaedia of Mathematics, M. Hazewinkel, Editor. 2001, Springer.

.56 Cottle, R.W., Pang, Jong-Shi, Stone, Richard E., The linear complementarity problem. Academic Press Inc., ed. C.S.a.S. Computing. 1992, Boston ,MA.

.57 Nayar, N.K.a.A.C.B.a.P.N.B.a.S.K., Attribute and Simile Classifiers for Face Verification, in In IEEE International Conference on Computer Vision (ICCV). 2009.

.58 PubFig, http://www.cs.columbia.edu/CAVE/databases/pubfig/.

.59 N. Kumar ,P.N.B., and S. K. Nayar, FaceTracer: A Search Engine for Large Collections of Images with Faces. European Conference on Computer Vision (ECCV), 2008: p. 340-353.

.60 OKAO, http://www.omron.com/r_d/technavi/vision.

.61 LibSVM, http://www.csie.ntu.edu.tw~/cjlin/libsvm./

.62 Guillaumin, M., Verbeek, J. and Schmid, C., Is that you? Metric learning approaches for face identification. Computer Vision, 2009 IEEE 12th International Conference, 2009 p. 498 - 505

.63 Kilian, Q.W. and K.S. Lawrence, Distance Metric Learning for Large Margin Nearest Neighbor Classification. J. Mach. Learn. Res., 2009. 10: p. 207-244.

.64 Jason, V.D., et al., Information-theoretic metric learning, in Proceedings of the 24th international conference on Machine learning. 2007, ACM: Corvalis, Oregon.

.65 Navneet, D. and T. Bill, Histograms of Oriented Gradients for Human Detection, in Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01. 2005, IEEE Computer Society.

.66 cited; Available from: http://vis-www.cs.umass.edu/lfw/results.html.

.67 Yaniv Taigman, L.W., and Tal Hassner, Multiple One-Shots for Utilizing Class Label Information. British Machine Vision Conference (BMVC), 2009.

.68 Bai, H.V.N.a.L ,.Cosine Similarity Metric Learning for Face Verification. Asian Conference on Computer Vision (ACCV), 2010.

.69 The Open University, Research day 2010 http://openu.ac.il/research/invitation2010.html

.70 Face.com, http://face.com

.71 Face.com, "Developers", http://developers.face.com/tools/#faces/detect

Abstract

While face recognition is a trivial task for the human brain, it still remains a complicated and challenging task for computer vision algorithms. This is because a digital representation of an image is highly influenced with changes in human face such as different poses, lightning, glasses, beard or a mustache, facial expressions, etc.

In recent years, face recognition research has attracted much attention both in the academia and in the industry. Face Recognition Technology (FRT) is applied today in biometric techniques, in security systems and in applications that operate on image databases on the internet (search engines, social networks). The rising interest in this field led to a great improvement in face recognition algorithms over the past 20 years, and lately many experiments show that these algorithms are becoming closer and closer to human face recognition abilities. Yet, a significant performance gap still exists, especially in unconstrained images, that is in images that show a large range of variation in pose, lightning, focus, resolution, facial expression etc.

This work deals with a sub-problem in the field of face recognition, pair-matching in unconstrained images, which is defined as follows: given two unconstrained images, each of which contains a face, decide whether the two people pictured represent the same individual. Here we review several state-of-the-art machine-learning based methods aimed at solving the pair-matching in unconstrained images problem, achieving amongst the best known results reported on the Labeled Face in the Wild (LFW), a database containing images downloaded from Yahoo! news website and show large variety in pose, expression, lightning etc.

Table of Contents

The Open University of Israel

Department of Mathematics and Computer Science

Face-Matching in Unconstrained Images

Final Paper submitted as partial fulfillment of the requirements

For an M.Sc. degree in Computer Science

The Open University of Israel

Computer Science Division

By

Tal Agmon

Prepared under the supervision of Dr. Tal Hassner

March 2011