האוניברסיטה הפתוחה
המחלקה למתמטיקה ולמדעי המחשב

שיטות לסיווג מידע ויזואלי באמצעות קידוד דליל ומקומי

עבודה מסכמת זו הוגשה כחלק מהדרישות לקבלת תואר
"מוסמך למדעים" M.Sc. במדעי המחשב
באוניברסיטה הפתוחה
החטיבה למדעי המחשב

על-ידי
אייל דהרי

העבודה הוכנה בהדרכתו של ד"ר טל הסנר

פברואר 14

תוכן עניינים

רשימת טבלאות

רשימת איורים

רשימת נוסחאות

תקציר

בעיית סיווג אובייקטים בתמונה הינה אחת הבעיות המאתגרות בתחום הראייה הממוחשבת. הקושי טמון בצורך להבחין ברב צורתיות ומראה של אובייקטים שונים השייכים לאותה מחלקה ובו בעת להימנע מאבחנות שגויות. למשל, סיווג חתול למחלקת הכלבים היא שגיאה. אמנם חתולים וכלבים חולקים לא מעט תכונות כמו זנב, הליכה על ארבע, שפם, אוזניים מחודדות, פרווה וכד', אך המבנה והמרקם של כל תכונה, על אף הדמיון הרב, הם כידוע שונים. בנוסף, ישנו צורך להבחין בשוני תוך מחלקתי – אוזניים של כלב מזן פודל שונות מאילו של כלב מזן לברדור. לפיכך, שיטת סיווג טובה אמורה להבחין בין מגוון רחב של אובייקטים שונים השייכים לאותה מחלקה ובאותו הזמן לא לבלבלם עם אובייקטים כמעט זהים ממחלקה אחרת.

ישנן עוד מספר בעיות המקשות על פעולת הסיווג כגון איכות הצילום - שינויי תאורה או עיוות. הסתרה - אדם המצולם מאחורי עצם המסתיר את פלג גופו העליון. רקע "רועש" - background clutter - צילום של זיקית המסווה את עצמה על רקע של גזע עץ עליו היא מטפסת וכד'. בעיות כגון אלו מקשות על הסיווג ודורשות שיטות ופתרונות מסוגים שונים (איור 1).

איור 1 – דוגמאות לסוגי בעיות המקשות על פעולת סיווג

מימין לשמאל: רקע רועש, שונות תוך מחלקתית, הסתרה, עיוות.

יתר על כן, סיווג יעיל בדרך כלל נמדד על ידי כמה פרמטרים: איכות הדיוק, זמן בניית מסווג, זמן הסיווג בפועל, נפח של המידע המיוצג, נפח המודל המסווג ועוד. לעיתים קרובות נראה כי פרמטרים אלו באים האחד על חשבון השני וישנו צורך לבצע אופטימיזציה כדי לקבל תוצאות מיטביות כפי שניתן לראות בסעיף (9).

כיום, ישנם פרויקטים ויישומים רבים המשתמשים בטכנולוגיות לסיווג מידע ויזואלי. רכבים אוטונומיים המנווטים בעצמם לאורך קילומטרים של סביבה מורכבת, מזל״טים, לוויינים, אמצעי התרעה מפני מכשולים בדרך או תאונה ברכבים אזרחיים [1] ויישומי מציאות רבודה הם רק רשימה חלקית של פרויקטים ויישומים המשתמשים בטכנולוגיות לסיווג מידע ויזואלי.

אמנם ישנה התקדמות טכנולוגית רבה ומשמעותית בשנים האחרונות בתחום סיווג המידע הוויזואלי, אך עדיין ישנן בעיות רבות שמחכות לפתרון. בשנים האחרונות ישנו מחקר עצום, התעוררות הולכת וגוברת ופרויקטים רבים, לדוגמה [2], כדי למצוא שיטות חדשות ולשפר את השיטות הקיימות בתחום הראייה הממוחשבת בכלל ובתחום הסיווג בפרט. חלק מהבעיות ששיטות אלו מציגות ודרכים לפתרונן אתאר בעבודה זו.

1 מבוא

המטרה המרכזית בבעיית הייצוג והסיווג של מידע ויזואלי היא ייצוג של המידע בצורה דחוסה ששומרת את התכונות המקוריות שלו ומסווגת אותו למחלקה הרלוונטית (מטרות נלוות: בדיוק מקסימלי ובזמן סביר) [3]. כלומר, המטרה הפרקטית היא לענות על שאלה כמו: "מה אתה רואה בתמונה?" (תשובה אפשרית: חתול), בשונה מבעיות העוסקות באימות כמו - "האם זה חתול?", או בזיהוי - "האם זה החתול שמילי?", או בחיפוש - "האם יש חתול בתמונה?". לכל אחת מבעיות אלו קיימות שיטות ייחודיות לפתרון למרות שבמקרים רבים ישנה חפיפה בין הפתרונות השונים. למשל, 'שק מאפיינים' [4] היא שיטה לייצוג וסיווג תמונות באמצעות דיאגרמת מאפיינים מקומיים, אולם ישנם פתרונות שמשתמשים בה גם לזיהוי [5].

1.1 חשיבות

פעולת הסיווג הנה מרכיב יסודי בראייה האנושית [6]. בזמן נהיגה למשל, אנו לרוב מביטים למרחב הכביש שנפרש לפנינו באופן כללי ללא כוונת זיהוי או חיפוש פרטני. מה שמעניין אותנו זו המחלקה אליה שייכים האובייקטים אותם אנו רואים במרחב, ולא אופי האובייקטים עצמם. כלומר, בזמן נהיגה חשוב לנו שלא לדרוס הולך רגל או להתנגש במכונית. זה לא רלוונטי מבחינתנו, באופן עקרוני, מי הם אותם הולכי הרגל שאנו רואים או מהו דגם המכונית שלפנינו.
אותו צורך בדיוק קיים גם במהלך טיסה, בתפעול מערכות בקרה, מערכות אבטחה ועוד. לכן, ככל ששיטות לסיווג יהיו יעילות, מהירות ומדויקות יותר, נוכל לאמן מערכות אוטומטיות לסייע לנו במגוון רחב של תחומים בחיי היום יום. למעשה, בעתיד נוכל אולי לאמן מערכות אלו "לראות" במקומנו.

1.2 מטרת העבודה

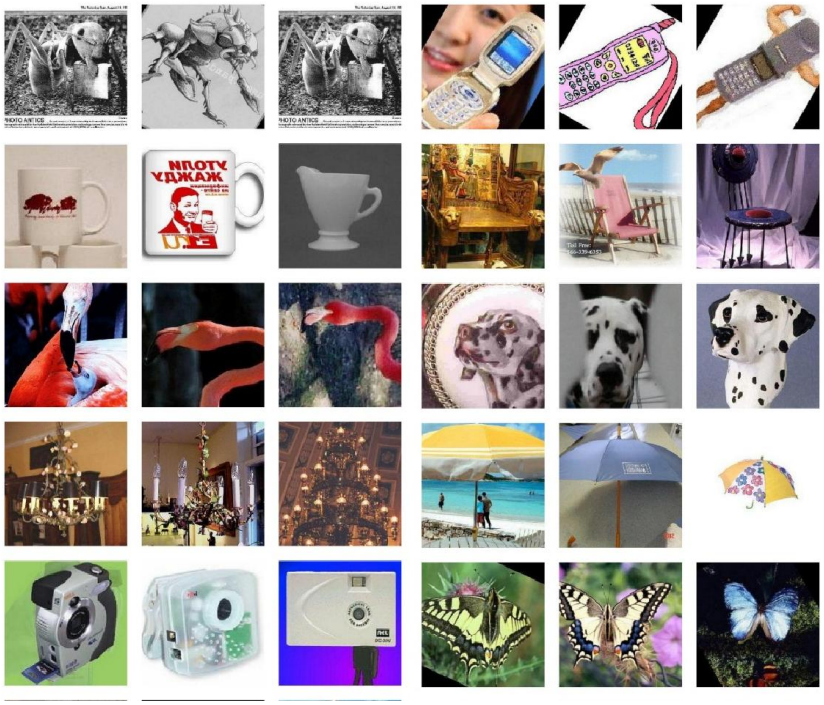
בעבודה זו אסקור מספר שיטות עכשוויות המשפרות ייצוגים של תמונה וביצועים של סיווג תמונות בעזרת קידוד דליל המתאימות לסביבות בהן יש צורך לטפל בכמות גדולה של מידע בזמן סביר וניצולת משאבים מצומצמת.
שיטות אלו נמצאו כמאוד יעילות לייצוג וסיווג מידע ויזואלי ובמאמרים בהם אתמקד אסקור כיצד הן משתלבות ומשפרות את השיטות 'שק מאפיינים' [4] ו- 'זיווג דמוי פירמידה מרחבי' [7]. שיטות אלו הראו ביצועים טובים על מספר מאגרי תמונות [8] [9] [10]. מאגרי תמונות אלו מכילים מספר רב של תמונות וטקסטורות, ברזולוציות שונות, בעלות מגוון רחב של תכונות כמו מחלקות שונות של אובייקטים ומקומות, שינויי זוויות צילום, קנה מידה, "רקע רועש" וכד'.

2 סיווג בקווים כלליים ומושגי יסוד

בחלק זה ארחיב מעט על האופן בו תמונה מיוצגת על ידי מחשב ועל הקושי בפעולת הסיווג.

מידע ויזואלי מיוצג על מחשב בצורה של מערך של פיקסלים. מידע זה מתורגם על ידי חומרה ותוכנה מתאימה לתמונה המוצגת על אמצעי תצוגה. מכאן שמחשב לא באמת "יודע" מה הוא המידע המוצג על אמצעי התצוגה. המחשב "מכיר" רק את הקידוד של המידע כדי שיוכל להציגו על אמצעי התצוגה.
כדי שמחשב יוכל לענות על שאלה כמו: "מהם האובייקטים המופיעים בתמונה?" עליו להשתמש באלגוריתמים ומניפולציות על המידע הוויזואלי שמיוצג על ידי מערכי פיקסלים במחשב.

2.1 "כל ילד יכול לעשות את זה" – כמה קשה לסיווג?



איור 2 – תמונות ממאגר התמונות Caltec 101

זו פעולה לא קלה לעין האנושית למצוא את המשותף לתמונות הללו וקטלג אותן. על אחת כמה וכמה למחשב.

באופן כללי, ניתן לומר שהשוני בין ייצוג תמונה - מערך פיקסלים אחד לאחר על מחשב טמון במערכים בעלי ערכים שונים של פיקסלים. כמו שניתן לראות ב (איור 2) , קשה לעיתים להבין מהו האובייקט המצולם מכיוון שלא תמיד האובייקט נמצא באותה זווית צילום, רזולוציה, סביבה, תאורה, איכות צילום ועוד. למעשה, קיימים אין סוף ייצוגים לאובייקט מסוים כמספר הערכים שיכול לקבל כל פיקסל או קבוצת פיקסלים המרכיבים תמונה והסביבה שלהם. כלומר, פעולת הסיווג שכל ילד קטן יודע לבצע בקלות היא פעולה מאוד מורכבת וקשה למחשב.

לעיתים קרובות, קשה מאוד להבין איך פעולה כל כך פשוטה ואינטואיטיבית שהעין והמוח האנושיים עושים נעשית מאוד מורכבת כאשר מדובר בחיקוי שלה על ידי מערכות ממוחשבות. למעשה קשה למצוא היום מערכת ממוחשבת או אלגוריתמים שמבצעים פעולה זו בצורה מושלמת. אבל, עם הזמן, ובמיוחד בשנים האחרונות, ישנה קפיצת מדרגה ושיפור ניכר בדיוק וביעילות של מערכות ממוחשבות ואלגוריתמים שמדמים את הראייה האנושית.

2.2 מספר מושגי יסוד

כמו שנראה בשיטות שאסקור [11] [12] אלגוריתם הסיווג מחולק לכמה שלבים עיקריים: למידה, הכללה והסקה.
לדוגמה, כדי לענות על שאלה כמו "מה אתה רואה בתמונה?", נראה לעיתים קרובות את השלבים הבאים:

1. שימוש בידע קודם (סט לימוד - training set של ייצוגים/דוגמאות) באמצעותו המערכת יוצרת הכללה - מודל חישובי שמייצג באופן כללי את האובייקטים שנלמדו. שלב זה נקרא גם שלב הלימוד [13].
2. כאשר יוצג למערכת אובייקט חדש ממחלקה שקיים מודל/אלגוריתם חישובי עבורה - מחלקה ש "נלמדה", נוכל לסווג אותו אליה על סמך הדמיון בינו לבינה. ישנן דרכים שונות להתאמה בין מידע מיוצג אחד לבין המודל של המחלקה אליה הוא שייך מסט הלימוד [14] [4]. שלב זה נקרא גם שלב הבדיקה - testing phase.

בעבודה זו נראה שתי צורות של למידה. אחת, תיוג אובייקטים הנקראת גם "למידה מונחית" (supervised learning) [15]. השנייה, הכללה על ידי מציאת תבניות או דפוסים נקראת גם "למידה לא מונחית" (unsupervised learning) [15].

בלמידה מונחית עבור כל אובייקט בקלט נתון הפלט המבוקש. כלומר, נתייג במערכת את המחלקה אליה שייכת כל דוגמת קלט ואז המערכת תייצר מודל של האובייקטים המתויגים. לדוגמה, כאשר רוצים ללמד את המערכת לסווג כלבים. "מתייגים" תמונות של כלבים כקלט והמערכת בונה מודל לסיווג מחלקת הכלבים.
בלמידה לא מונחית לא ידוע מהו המודל עבור קבוצת הייצוגים בזמן הלמידה. כתוצאה מכך, המערכת צריכה למצוא בעצמה דפוסים או תבניות בקבוצת הייצוגים הנלמדת ולתייג כל תבנית בעצמה. דוגמה לאלגוריתמים שמיישמים למידה לא מונחית ניתן למצוא בנספחים (ג) ו-(ד).

3 השיטות בהן עבודה זו מתמקדת

השיטה הראשונה בה אתמקד נקראת זיווג לינארי דמוי פירמידה מרחבי על בסיס קידוד דליל (ScSPM) Sparse coded SPM [11], היא הרחבה לשיטת סיווג המתבססת על מסווג (SVM) Support Vectors Machine (נספח א) עם גרעין זיווג דמוי פירמידה מרחבי (SPM) Spatial Pyramid Matching [16].

ובכן, SPM הנה שיטה מצליחה ופופולארית לסיווג מידע ויזואלי. אולם, מסווג ה SVM בו היא משתמשת הוא לא לינארי, מה שנותן, כפי שנראה בהמשך, סיבוכיות $O(n^2 \sim n^3)$ בשלב הלימוד ו $O(n)$ בשלב הבדיקה כאשר n הוא גודל סט האימון.
מכך ניתן להסיק שלא יהיה זה יעיל חישובית לטפל בסט לימוד המכיל מעבר לאלפי תמונות. השיטה ScSPM אותה אסקור משתמשת במסווג SVM לינארי המבוסס על קידוד דליל (SC) Sparse Coding סעיף (6) של מתארי Scale-Invariant Feature Transform (SIFT) סעיף (5.1.1). שיטה חדשנית זו משפרת בצורה משמעותית את הסיבוכיות בשלב הלימוד לכדי $O(n)$ וסיבוכיות קבועה בשלב הבדיקה.

השיטה השנייה בה אתמקד נקראת קידוד לינארי מקומי (LLC) Locality-constrained Linear Coding for Image Classification [12], מציעה גם היא שיפור ל SPM המבוסס על שק מאפיינים Bag of Features (BoF) סעיף (5.1). השיפור בא לידי ביטוי בשלב קידוד (VQ) Vector Quantization מתארי ה SIFT של השיטה. במקום VQ של מתארי SIFT, LLC מקודדת ומשתמשת במיקום של כל מתאר SIFT כדי להטיל אותו למערכת קואורדינטות לפי מיקומו במרחב. לאחר מכן, כל הקואורדינטות המוטלות משולבות באמצעות Max Pooling כדי ליצור את הייצוג של המידע המסווג. LLC בשיתוף עם מסווג לינארי מציגה ביצועים יוצאי דופן אותם אנתח ואציג בעבודה.

4 מבנה העבודה

הפרקים של עבודה זו מאורגנים באופן הבא: בפרק 5 אסקור שיטות שמהוות בסיס למנגנון הסיווג של ScSPM ו LLC. בפרקים 6 ו 8 אתמקד בשיטות ScSPM ו LLC, איך הן משפרות את ביצועי השיטות BoF ו SPM ואציג את הביצועים והיעילות שלהן על מאגרי נתונים שונים של תמונות כמו Caltech-101. בפרק 9 אציג את תוצאות העבודות ואשווה בין תוצאות של שיטות שונות ובמיוחד בין השיטות ScSPM ו LLC. בפרק 10 אדגים אפליקציה שמשתמשת ב LLC לסיווג. בפרק 11 אציג מסקנות ואפשרויות לפיתוח עתידי.

5 סקירה של שיטות

השיטה הראשונה אותה אסקור נקראת שק מאפיינים (Bag of Features (BoF. אך בספרות ניתן גם למצוא שמות מקבילים כמו Bag of Words (BoW) או Bag of Keypoints (BoK) כאשר המילה שק (Bag) לעיתים קרובות מוחלפת בהיסטוגרמה (Histogram). מונחים אלו מופיעים לחלופין בספרות כדי לתאר את מודל BoF המכיל אוסף מילים המתארות אובייקטים ויזואליים המסודרים במילון. חלקים רבים של מודל BoF משמשים את השיטות שבהן אתמקד בעבודה זו ולכן אסביר את BoF בפירוט ובהמשך אראה איך השלבים הרלוונטיים של השיטה משתלבים בשיטות שאסקור.

5.1 שק מאפיינים

בשיטה BoF תמונות מיוצגות על ידי אוסף לא סדור של תכונות מקומיות. השם במקור מגיע מייצוג Bag of Words בו משתמשים לחילוץ מידע טקסטואלי בתחום הנקרא "עיבוד שפה טבעית" Natural Language Processing (NLP) [13]. בתחום זה מסמכים מיוצגים על ידי אוסף לא סדור של מילים משמעותיות . לדוגמה, מסמך המכיל מילים כמו: גרמניה, בעלות הברית, נאצים, סובייטים וכד'. בצורה טבעית יש סיכוי מאוד סביר שיעסוק במלחמת העולם השנייה (איור 3). כלומר, מסמך המכיל אלפי מילים יכול להיות מיוצג רק על ידי מספר מילים מועטות וילמד הרבה על תוכנו הכולל.

באותה צורה ניתן להסתכל על תמונה כעל מסמך טקסטואלי שהמילים בו מיוצגות כמידע ויזואלי. אחת העבודות הראשונות בה השתמשו ברעיון זה היא [4]. על סמך הקבלה זו, בעבודות רבות [17] [7] [4] ניתן למצוא מושגים שקולים כמו:

code vector ~ visual word ~ word

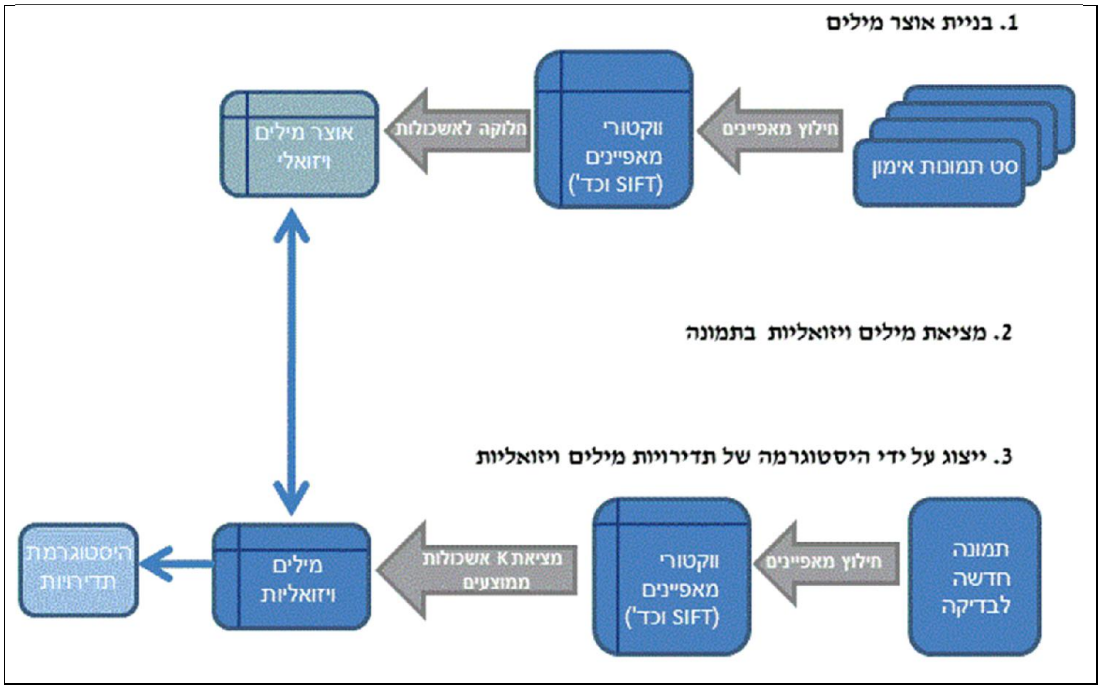
codebook ~ visual vocabulary ~ dictionary

איור 3 – ענן מילים

ספירת תדירויות של מילים משמעותיות המייצגות מסמך שלם והצגתן כענן מילים כאשר גודל המילה מעיד על כמות המופעים שלה בטקסט.

כאמור, המונח BoK (Bag of Keywords) נקבע לראשונה על ידי [4]. צוות זה היה בין הראשונים שהשתמשו והדגימו שיטה זו לפתרון בעיות סיווג. במודל הראשוני שלהם לפתרון הבעיה הם השתמשו במתארי SIFT של שטחים אפינים כדי לייצג נקודות מפתח בתמונות של כל מחלקה שנלמדה. לאחר מכן, הריצו אלגוריתם k-means (נספח ד) כדי לבנות אוצר מילים ויזואלי של אובייקטים והשתמשו במסווג SVN כדי לסווג אובייקטים למחלקות שלהם. לפי התוצאות שהוצגו במחקר מקיף מסווג ה SVM הראה דיוק רב יחסית למסווגים אחרים.

השיטה ברמת התהליך מבוססת על למידה בכמה שלבים (איור 4). בשלב האימון יוצרים אוצר מילים ויזואלי של תמונות של אובייקטים מתוך סט הלימוד. כל מילה באוצר מילים זה מיוצגת על ידי היסטוגרמה של מתארי SIFT. כלומר, כל היסטוגרמה כזו סופרת בכל תא כמה מתארי SIFT ישנם אשר מתארים את אותו המידע. לדוגמה, אם המידע הנלמד הוא פנים של אנשים, אז ההיסטוגרמה שמייצגת פנים תהיה מורכבת מתאים של מאפיינים מקומיים של פנים. מאפיינים כמו : אף, עין, אוזן, פה, שיער וכדי. כל תא בהיסטוגרמה סוכם את הכמות של כל מאפיין של פנים שהופקה מהתמונה.



איור 4 – תיאור תהליך BoF בקווים כלליים

שלב הבדיקה מחולק לשניים. בשלב הראשון, תמונות הקלט מיוצגות על ידי היסטוגרמות של מתארי SIFT, בדיוק כמו בשלב האימון. בשלב השני, השיטה משתמשת ב SVM כדי לסווג את ייצוגי התמונות. ההנחה באופן כללי, שהיסטוגרמה שתכיל יותר מאפיינים של אף, עין, פה, אוזן ושיער תסווג למחלקת הפנים של אנשים. לעומת זאת, היסטוגרמה שתכיל יותר מאפיינים של גג, ארובה וחלונות תסווג למחלקת הבתים.

שיטה זו נבדלת משיטות אחרות בכמה אופנים עיקריים:

- היא לא מייחסת חשיבות למיקום הגיאומטרי של המאפיינים בתמונה. כל תמונה, כאמור, מיוצגת על ידי אוסף של מתארי SIFT (מכאן השם "שק מאפיינים").

- השיטה מייצגת כל תמונה על ידי היסטוגרמה של תדירויות של המתארים המקומיים שלה. זאת בניגוד לשיטות אחרות המשתמשות בכל המידע שנראה בתמונה, כיווני התכונות, הרזולוציות וכד'. ייצוג דליל זה של המידע הוא מאוד יעיל. לצורך המחשה, מערכת BoF יכולה להחזיק אוצר מילים ויזואליות בעל 100,000 ערכים ולחלץ כ-5,000 תכונות מתמונה בודדת. מכאן שבייצוג BoF ערכם של כמעט 95% מהתאים של כל היסטוגרמה שווה לאפס. תכונה זו עוזרת בנוסף לאחסון ושליפה של המידע כמו שנדון בהמשך.

- חיסכון בפעולות של "ניקיון" התמונה מרעשים. בשיטות לסיווג פנים למשל, ישנו שלב שאחראי על הפרדת הרקע של התמונה מהפנים [18]. ל BoF אין כל שלב כזה מכיוון שרקע

הוא אזור בעל חשיבות נמוכה מבחינת המידע שהוא מוסיף לתמונה. מתארי התכונות של השיטה מחלצים מעט מאוד מהמידע הזה.

מאז פורסמה השיטה ב 2004 ועד לאחרונה נעשה מחקר רב על האלגוריתם ונכתבו שיפורים רבים לשלבים המרכיבים אותו. חלקם שיפורים של איכות הדיוק בסיווג [19], חלקם שיפורי ביצועים וביניהם האצה של שלבי הלמידה השונים [20] [16] , צמצום נפח המידע המיוצג והאופן בו הוא מיוצג [11]. בסעיפים הבאים נעבור על מספר עבודות שנעשו בתחום המראות שיפור לשלבים השונים של השיטה וננתח את הצורה בה כל שלב פועל.

5.1.1 הפקת מאפיינים

כאמור, BoF מתבססת על מאפיינים שאינם משתנים והמרחבים שלהם. ברוב המחקרים השיטה משתמשת במאפייני SIFT אך ישנן גם עבודות שמשתמשות במאפיינים אחרים כפי שנראה בהמשך.

מאפייני Scale-Invariant Feature Transform (SIFT)

אחת השיטות הנפוצות ביותר כיום להפקת מאפיינים נקראת מאפייני SIFT [21]. השיטה נסמכת על מקומות בתמונה בהם יש שינוי בעוצמה של אור כמו פינה של שולחן או קווי מתאר של אובייקט. מקומות אלו נקראים "נקודות מפתח" (Keypoint) בתמונה. נקודות מפתח כאלו הוכחו כחסינות לטרנספורמציות אפיניות.

שיטה זו מחולקת לשני שלבים עיקריים.

בשלב הראשון, מאתרים נקודות מפתח בתמונה (Keypoint detection) עבורם מאתר SIFT מחושב (ישנן שיטות רבות למציאת נקודות מפתח בתמונה [22]). נקודות מפתח עוזרות לסיווג בכך שהן ייחודיות, מקומיות בתוך התמונה וקטנות, חסינות להסתרה במופעים שונים של אובייקט מסוים וחסינות לטרנספורמציות אפיניות.

בשלב השני, יוצרים מתאר ייחודי לנקודות המפתח (Keypoint descriptor) שנמצאו בתחילה ברזולוציות שונות. מתאר ייחודי זה חסין במידה מסוימת לטרנספורמציות פרויקטיביות, שינויי תאורה, עיוות וכד'.

שיטות ויישומים רבים כגון [23] [4] ובכללן השיטות אותן אני סוקר בעבודה זו משתמשים במאפייני SIFT כחלק מפעולתם.

מאפיינים שאינם משתנים במופעים שונים של אובייקט משמשים כמעיין חתימה של האובייקט. בעזרת אוסף חתימות ייחודיות אלו ניתן להבחין בין אובייקט לאובייקט בצורה מדויקת יותר. נמנה כמה מהתכונות שבשלן מאפייני SIFT חסינים כל כך:

1. המאפיינים בדרך כלל נאספים מאזורים בעלי ניגודיות גבוהה בתמונה. לדוגמה, פינות של אובייקט הנמצאות על קווי המתאר שלו המבדילים אותו מהרקע עליו הוא מצולם.

2. המיקום היחסי של המאפיינים לא משתנה בין תמונה לתמונה. למשל, אם נשתמש בארבע הפינות של חלון כמאפיינים. מיקומן לא ישתנה לא משנה איזו באיזו תמונה של חלון נסתכל. אבל אם נשתמש בכל הנקודות על דפנות החלון כמאפיינים, הזיהוי של החלון יפגע מכיוון שישנו הבדל בין מאפייני הנקודות הללו בין חלון פתוח לסגור למשל.

לפי Low [21] השלבים העיקריים לחישוב מאפייני ה SIFT הם כדלהלן:

1. איתור ברזולוציות שונות (Scale-space extrema detection). בשלב זה בוחנים את כל התמונה ברזולוציות ומיקומים שונים בצורה יעילה המשתמשת בפילטר שינוי גאוסיאני (DoG - difference-of-gaussian) כדי לאתר היתכנות לנקודות מפתח שהן חסינות לשינויי רזולוציה ומיקום.

2. יצירת מודל מקומי של נקודות מפתח (Keypoints localization). לכל מקום בתמונה בו ישנה היתכנות לנקודות מפתח מתאימים מודל מפורט לקביעת המיקום והרזולוציה. כל נקודה משמעותית כזו, נקבעת לפי מידת היציבות שלה ברזולוציות והמיקומים השונים.

3. התאמת כיוון (Orientation assignment). כיוון אחד או יותר מוקצים עבור כל מיקום נקודה מהשלב הקודם על ידי חישוב כיווני הגרדיאנט המקומי בתמונה.

אחרי השלבים 1, 2 ו-3. המידע מהתמונה שבידינו עבר התמרה יחסית להקצאת הכיוון, הרזולוציה, והמיקום של כל תכונה, ולכן מספק חסינות בפני הטרנספורמציות הללו.

4. מתאר נקודת מפתח (Keypoint descriptor). מודדים את השינויים המקומיים בגרדיאנטים בתמונה לרזולוציה מסוימת באזור מסביב לכל נקודת מפתח. שינויים אלו משמשים לייצוג המאפשר חסינות לרמות משמעותיות של עיוות צורה ושינויי תאורה מקומיים.

מתארי SIFT מיוצגים על ידי וקטורים מממד 128 המחושבים באופן הבא:

מתוך אזור דגימה בגודל 16×16 מסביב לנקודת מפתח לוקחים מערך היסטוגרמות כיוון של סביבה של כל פיקסל בגודל 4×4 עם 8 תאים, אחד לכל כיוון. המתאר הסופי הוא וקטור של כל הערכים של היסטוגרמות אלו. כיוון שההיסטוגרמות בגודל $4 \times 4 = 16$ ולכל אחת תאים בגודל 8 הגודל של כל מתאר SIFT הוא 128. הווקטור הזה מנורמל לגודל יחידה בכדי להיות יותר עמיד לשינויים אפינים בתאורה. בכדי לבטל עד כמה שאפשר שינויים לא לינאריים בתאורה משתמשים בסף תאורה בעוצמה מסוימת ואז מנרמלים מחדש.

מתארי HoG (Histogram of Oriented Gradients)

מתארי HoG [24] [25] מיועדים לתאר אובייקטים באזורים מקומיים בתמונה על ידי התפלגות של וקטורים של שינויים בעוצמות. מתארים אלו נוצרים בכמה שלבים.

בשלב הראשון, מחשבים לכל תא בתמונה את תמונת הגרדיאנט שלו על ידי מעבר על כל הפיקסלים ועבור כל פיקסל בודקים את השינוי לאורך ציר X לעומת השינוי לאורך ציר Y בערך של הפיקסל לכל כיוון. הערך הגדול ביותר לכיוון מסוים הוא הווקטור שהפיקסל יקבל בתמונת הגרדיאנט החדשה שתיווצר עבור התא.

בשלב השני, לכל תא בתמונה מתאימים היסטוגרמה של כיווני גרדיאנטים. כל תא בהיסטוגרמה מהווה טווח כיוונים. לדוגמה, תא שמייצג את גדלי הווקטורים בטווח הכיוונים $90°$ עד $120°$ ותא שמייצג את כל גדלי הווקטורים בטווח הכיוונים $121°$ עד $150°$. כלומר, מכל תא בתמונה אוספים את כיווני הווקטורים בסביבה שלו וגודלו של כל וקטור מכיוון מסוים נסכם בתא המתאים לכיוון הרלוונטי בהיסטוגרמה.

בשלב השלישי, מנרמלים את התאים על ידי חלוקה של התמונה לקבוצות תאים – בלוקים. בכל בלוק מאחדים את כל ההיסטוגרמות לווקטור אחד ולפי ייצוג היסטוגרמה זה הנקרא "אנרגיה" מנרמלים את כל התאים בבלוק.

לסיום, קיבלנו ייצוג וקטורי (היסטוגרמה של גרדיאנטים מכוונים) שמתארת כל בלוק בתמונה. בעבודה LLC אותה אסקור בפרק 8 משתמשים במתארי HoG כדי להגיע לתוצאות סיווג פורצות דרך בדיוק, מהירות וחיסכון בזיכרון.

שיטות לאיתור נקודות מפתח בתמונה

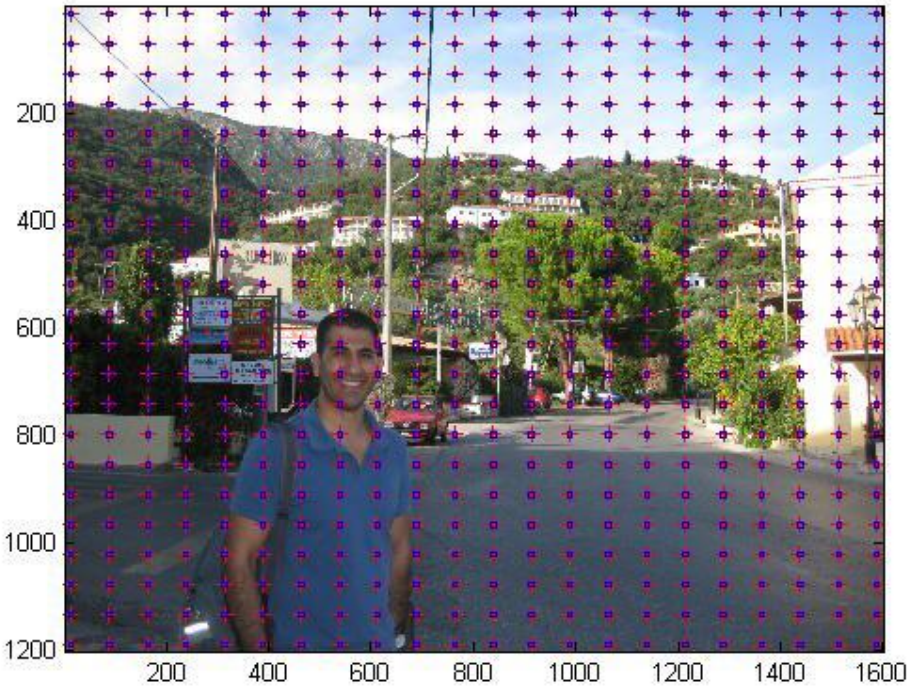
מכיוון שישנה חשיבות רבה לשלב מציאת נקודות המפתח בתמונה והשפעה ישירה על ביצועים, נרחיב מעט על שיטות לאיתור נקודות מפתח בתמונה.

כאמור, BoF בצורתה הבסיסית משתמשת ב פילטר DoG למציאת נקודות המפתח בתמונה. פילטר זה מייצג כל נקודת מפתח בעזרת שלוש קואורדינטות מקומיות (x, y, s) וערכי אקסטרמה של האזור התלת ממדי הנוצר מייצוג זה (הקואורדינטה השלישית מציינת את הרזולוציה). בעבודה [26] בנו מזהה נקודות מפתח לפי הרזולוציה הגדולה ביותר. נקודת מפתח במזהה שבנו מחזיקה אנטרופיה גבוהה בתוך מרחב רזולוציה מקומי.

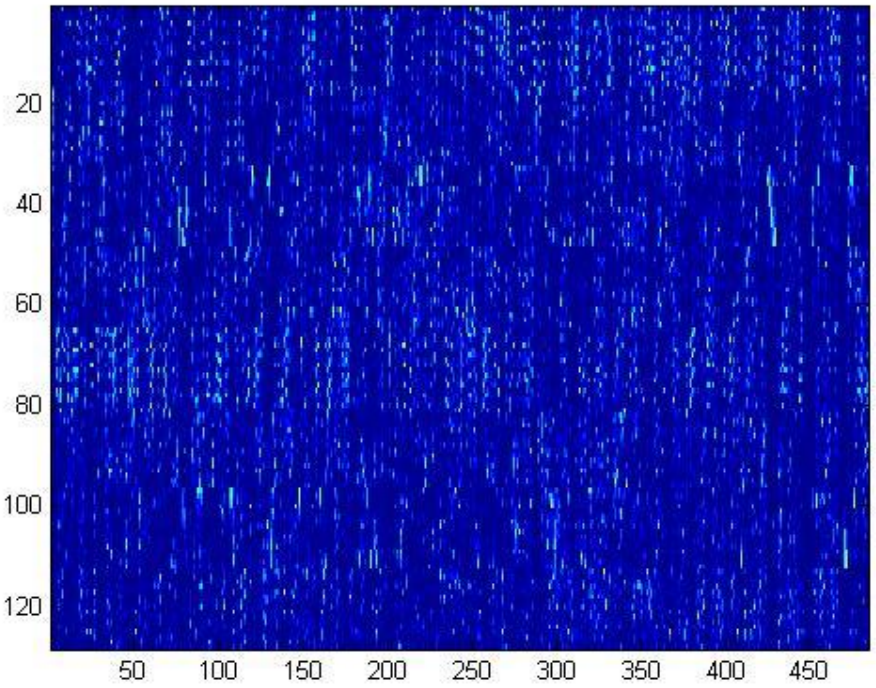
מזהה נקודות מפתח נוסף [27] נקרא Harris Affine Detector. מזהה זה מרחיב ומשפר את הייצוג של Harris ו Stephens [28] לייצוג מרחב רזולוציות עם אזורי כיוון אליפטיים.

מזהה Harris Affine Detector מנסה למצוא אזורי אקסטרמה עמידים (MSER) Maximally Stable Extremal Regions ומחפש אזורים אליפטיים בתמונה בעזרת תהליכי משנה. כל אזור כזה מסווג כנקודת מפתח.

עוד עבודה מעניינת בנושא נעשתה על ידי [20]. בעבודה זו הם סוקרים אופטימיזציה של שלבי BoF. בשלב איסוף המאפיינים הם הגיעו למסקנה כי מתארי SIFT המשתמשים בשלב זיהוי נקודות המפתח בדגימה דחוסה ורנדומלית של אזורים מקומיים בתמונה נותנים ביצועים מצוינים של סיווג יחסית לשיטות שמשתמשות בנרמול עוצמות תאורה של פיקסלים בגווני אפור למשל. הם מראים ביצועים טובים של שיטות דגימה רנדומלית העולים על ביצועים של שיטות מציאת נקודות מפתח כמו DoG. הם מראים שהמדד הקובע הוא מספר האזורים שנבדקים בתמונה. ולכן ככל שנדגום בצורה יעילה אזורים בתמונה עד לסף דחיסה מסוים כך משתפרים הביצועים (איור 5) (איור 6).



איור 5 – דוגמה למיקום מאפייני SIFT שנדגמו בצורה טבלאית דחוסה



איור 6 – דוגמה לייצוג מאפייני SIFT

המאפיינים באיור זה הם ייצוג מממד $128 = 8 \times 4 \times 4$ של איור 5

5.1.2 בנית אוצר מילים

מציאת אשכולות מאפיינים (Clusters) המתארים את אותו עצם והקידוד שלהם משמשים לבניית אוצר המילים הוויזואלי ב BoF. ישנם לא מעט אלגוריתמים למציאת אשכולות מאפיינים, נזכיר כמה: K-means (נספח ד), hierarchical clustering [29] ו- k-d trees [30].
לכל אחד יתרונות וחסרונות משלו הבאים לידי ביטוי במהירות, דיוק, מקום וכד'. בנוסף, כל אחד מתאים לסוג מסוים של פיזור מאפיינים. לעיתים רבות אותם אלגוריתמים למציאת אשכולות מאפיינים המשמשים אותנו בזמן הלמידה, משמשים גם לקידוד מילים באוצר המילים בזמן הבדיקה, על כך נפרט בהמשך.

ובכן, אחרי שהפקנו מאפיינים מסט הלימוד נשתמש במאפיינים אלו לבניית אוצר המילים. ניקח את K-means כדוגמה לבניית אוצר המילים. מריצים את האלגוריתם מספר פעמים עד להתכנסות - שלב בו אין שינוי גדול בין ממוצע אחד למשנהו בכל אשכול. כל ממוצע כזה נספר בתאים של היסטוגרמת מאפיינים.

היסטוגרמה של מאפיינים, הנקראת גם "מילה" "במילון" של היסטוגרמות מאפיינים, היא למעשה מערך של תאים. כל תא מתאר חלק אחר מהאובייקט אותו אנחנו לומדים. אחרי שקיבלנו מכל

תמונה של האובייקט את סט הממוצעים שלה לוקחים את הממוצעים שנלמדו ובודקים לאיזה תא כל אחד מתאים. בכל תא סוכמים את מספר הממוצעים ומקבלים היסטוגרמה של תדירויות ממוצעים.

5.1.3 התאמה בין דיאגרמות מאפיינים

עד כה הסברתי איך בונים ייצוג לתמונה בשיטת BoF. בסעיף זה אפרט על בניית מודל שמאפשר סיווג של ייצוג זה למחלקה.

הערה: כאשר רוצים לסווג אובייקט מסוים למחלקה אליה הוא שייך בשיטת BoF, מבצעים פעולת קידוד מתארי SIFT. פעולה זו נקראת בספרות Quantization.

נתחיל בפונקציות למדידת התאמה בין היסטוגרמות של מאפיינים. פונקציות אלו נקראות פונקציות גרעין [31]. הפונקציות הללו מיושמות בגרעין של מכונת וקטורים תומכים (SVM) בעזרתן האלגוריתם מחשב התאמה בין אבייקט למחלקה אליה הוא שייך. מכונת וקטורים תומכים שייכת לתחום מסווגים הנקראים 'מסווגים מבחינים' (Discriminant Classifiers). מסווגים אלו נמצאו כיעיל לעבודה עם דיאגרמות מאפיינים [20].

בעבודה [32] משווים מספר פונקציות גרעין למציאת התאמה אופטימלית בין היסטוגרמות מאפיינים. הנה מספר פונקציות מרחק שנבחנו:

- **מרחק L1**

(1.5) – מרחק L1

$$D(h_1, h_2) = \sum_{i=1}^{N} |h_1(i) - h_2(i)|$$

מרחק זה הוא מקרה פרטי של Minkovski-Form-Distance המחשב סכום של מרחקים מוחלטים.

- **מרחק $\chi 2$**

(2.5) – מרחק 'כי'

$$D(h_1,h_2) = \sum_{i=1}^{N} \frac{(h_1}{h_1} \frac{(i) - h_2}{(i) + h_2} \frac{(i))^2}{(i)}$$

מחשב מרחק ומנרמל בגודל התאים שהוא מספר המאפיינים המקומיים שנמצאו המתארים את אותו נושא. הנרמול נעשה בכדי לתת משקל יחסי לכל המאפיין מקומי ובכך להימנע מסטיית תקן של כמות גדולה של מאפיין מקומי מסוים אשר משפיע על המרחק הכולל בין שתי היסטוגרמות.

- **מרחק ריבועי (Quadratic distance)**

(3.5) – מרחק ריבועי

$$D(h_1,h_2) = \sum_{i,j} I_{ij}(h_1(i) - h_2(j))^2$$

לתת מראש משקל יותר גבוה למאפיינים מסוימים על ידי הכפלה במטריצת זהות כדי למצוא מרחק של דמיון בין מאפיינים. ובכך להימנע ממצב בו ההשוואה בין מאפיינים תהיה רק מרחבית.

ישנן פונקציות גרעין שמשתמשות בפונקציות המרחק שתיארנו לעיל, לדוגמה:

- **ההכללה של גרעין גאוסיאני**:

(4.5) – הכללה של גרעין גאוסיאני

$$K(h_1,h_2) = \exp\left(-\frac{1}{A}D(h_1,h_2)^2\right)$$

משתמשים בגרעין זה כדי לנרמל את המרחק בין שתי היסטוגרמות כך שמרחקים גדולים מדי לא יקבלו יותר משקל. בגרעין זה, ניתן להחליף את D בכל אחת מפונקציות המרחק לעיל.

- **גרעין חיתוך היסטוגרמות**:

(5.5) – גרעין חיתוך היסטוגרמות

$$I(h_1,h_2) = \sum_{i=1}^{N} \min(h_1(i), h_2(i))$$

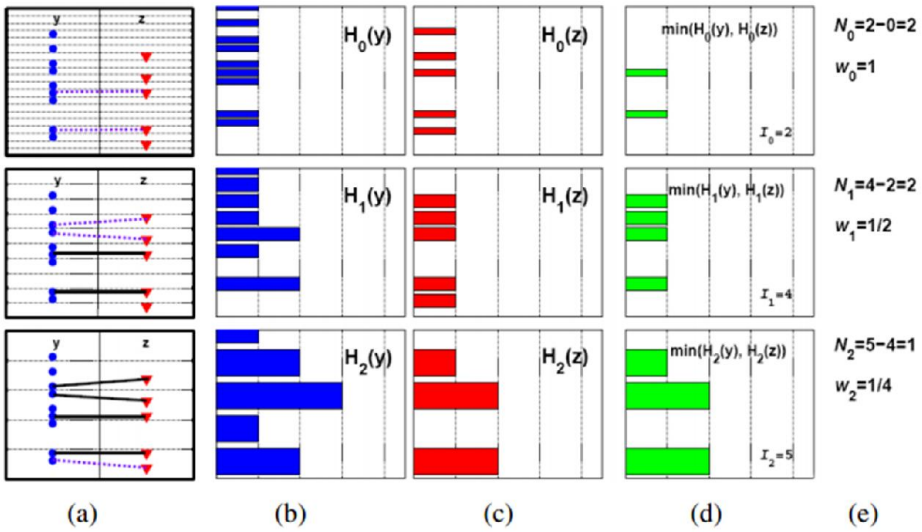
משתמשים בגרעין זה בדרך כלל כאשר עובדים על היסטוגרמות לא מאותו סדר. כלומר, כאשר היסטוגרמה אחת יותר ארוכה מהשנייה. ניתן לראות שגרעין זה הוא הכללה של L1. נשתמש בגרעין חיתוך היסטוגרמות כאשר נסביר את הפעולה של גרעין SPM המשתמש בדרוג סכום חיתוך היסטוגרמות לפי משקלים ברזולוציות שונות [7] [16].

עד כה עברנו על השלבים העיקריים של המודל. לסיכום: בשלב הלמידה, עוברים על סט תמונות של מספר אובייקטים ומפיקים מהן מאפיינים כפי שמתואר בסעיף (5.1.1). לאחר מכן בונים את אוצר המילים כפי שמתואר בסעיף (5.1.2). אוצר מילים זה מועבר כקלט מתויג ל SVM. בזמן הבדיקה, כדי לסווג אובייקט חדש X, נבצע עליו את השלבים (5.1.1) ו- (5.1.2). נעביר את ייצוג ההיסטוגרמה של האובייקט X ל SVM ונקבל את הסיווג של X לפי ההתאמה למילה במילון שבנינו בשלב הלמידה.

5.2 זיווג דמוי פירמידה מרחבי

כאמור, בצורתו הבסיסית ייצוג BoF לא מכיל מידע על המיקום הגיאומטרי של המאפיינים בתמונה. כל תמונה מיוצגת על ידי אוסף תדירויות של מתארי SIFT. אמנם צורה זו של ייצוג מאוד חסכונית ויעילה אך במקרים מסוימים ישנו יתרון רב לייצוג המתחשב גם במיקום המרחבי של המאפיינים בתמונה. דוגמה לכך היא בעיות סיווג נוף, סיווג מקום או סיווג אובייקט מסוים. בבעיות אלו, למידע המרחבי יש יתרון משמעותי והוא תורם בצורה משמעותית לדיוק הסיווג. דוגמאות לסיווג נוף: מדבר, ים וכד'. דוגמאות לסיווג מקום: משרד, רחוב וכד'. דוגמאות לסיווג אובייקט מסוים: בני אדם, עצים וכד'.

בעבודה [16] מציעים פונקציית גרעין זיווג דמוי פירמידה מרחבי Spatial Pyramid Matching (SPM) אשר מאפשר התחשבות במיקום המרחבי של המאפיינים בתמונה. בנוסף, הם מראים שכאשר הם משתמשים בגרעין זה במערכת שלהם, היעילות של הלמידה היא לינארית כתלות במספר המאפיינים מבלי לפגוע בדיוק בהשוואה לשיטות אחרות שיעילותן היא פולינומיאלית כתלות במספר המאפיינים.



איור 7 – דוגמה מופשטת מממד 1 להתאמה דמוית פירמידה

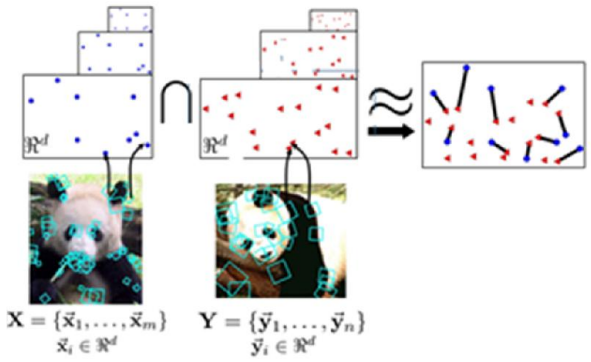
ההתאמה של מאפיינים נקבעת כאשר הם נמצאים באותו התא של ההיסטוגרמה. בדוגמה לעיל שני סטים של מאפיינים יוצרים שתי פירמידות של היסטוגרמות. כל שורה מתאימה לשלב אחר. כמו שניתן לראות השלב הראשון לא מיוצג בדוגמה מכיוון שאין התאמה של מאפיינים בשלב זה. ב (a), הסט y בצד השמאלי והסט z בימני. המאפיינים מפוזרים במאוזן ואותם מאפיינים חוזרים על עצמם בכל שלב. הגבולות בין התאים מיוצגים על ידי הקווים המנוקדים הבהירים. התאמה בין מאפיינים בשלב מסוים מיוצגת על ידי הקווים הבולטים כאשר הם נופלים באותו התא. ב (b) ו (c), רואים היסטוגרמות ברזולוציות שונות עם מספר המאפיינים שלהן. ב (d) רואים את תוצאת חיתוך ההיסטוגרמות (b) ו (c). ב (e) רואים את תוצאות ההתאמה בין ההיסטוגרמות בכל שלב שנמדדות על ידי משקל שונה לכל שלב. התאמה בשלב בו הרזולוציה גבוהה תקבל תוצאת משקל יותר גבוהה.

השיטה מחשבת בצורה מהירה את ההתאמה בין שתי דיאגרמות מאפיינים מממדים שונים. במקום לבצע שלב התאמה בדומה ל - 5.1.3, מחלקים את התמונה לרזולוציות שונות:

$2^l \times 2^l$ כך ש $l = 0,1,2,..$ (איור 8). מכל תא ברזולוציה מסוימת מרכיבים היסטוגרמה שסופרת את תדירויות וקטורי המאפיינים המתאימים בו (איור 7 a-c). יש לשים לב לכך שחלוקה כזו של התמונה לרזולוציות שונות מוסיפה למודל שמייצג אותה תלות במיקום המרחבי של המאפיינים שלה. המרחק בין שתי היסטוגרמות של מאפיינים שנוצרו בצורה כזו מחושב על ידי גרעין חיתוך היסטוגרמות (5.5) בין שני תאים מתאימים (איור 7 d). בנוסף, סוכמים את המאפיינים בכל שלב עם משקלים (איור 7 e) כאשר התאמות שנמצאו בשלבי רזולוציה גבוהה יותר מקבלים משקל גבוהה יותר. זאת מכיוון שאם שני מאפיינים נמצאים באותו התא ברזולוציה גבוהה זה מעיד על התאמה מרחבית גבוהה ביניהם (6.5). לדוגמה, מאפיינים של שמים יהיו בדרך כלל באזור העליון של התמונה, התאמה בין שני מאפיינים שמייצגים שמים באזור העליון של התמונה תקבל משקל יותר גבוהה מאשר שני מאפיינים כאלו ברזולוציה נמוכה יותר.

(6.5) – גרעין SPM

$$D_\Delta = \sum_i w_i N_i \,, \qquad N_i = C_i - C_{i-1} \; and \; w_1 = \frac{1}{d2^i}$$

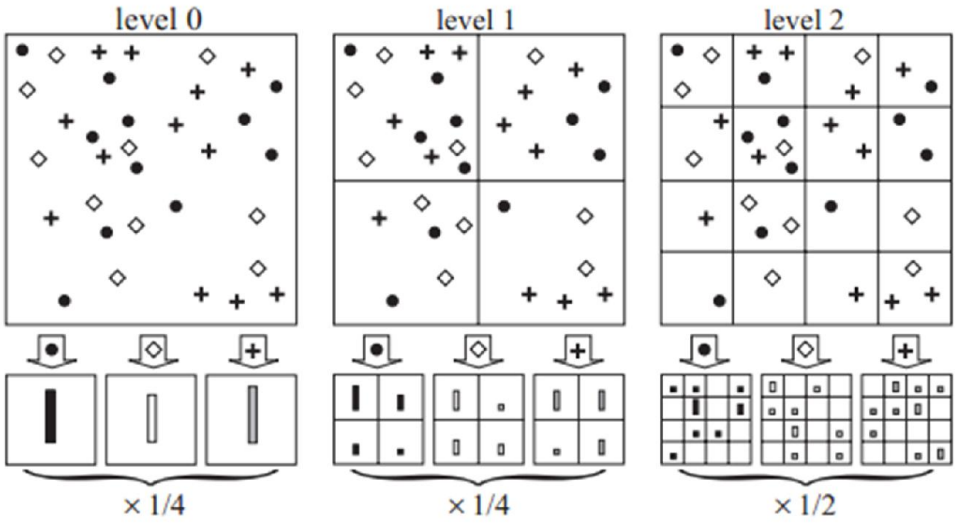


איור 8 – דוגמת המחשה לגרעין SPM בעבודה [16]

בנייה של פירמידת התאמה מרחבית מקבוצות מאפיינים בתמונה.

בעבודה [7] מציעים הרחבה ל BoF הדומה לרעיון העומד מאחורי העבודה [16]. השיטה שלהם מציעה מאפייני SPM מבוססי היסטוגרמות. שלב הפקת המאפיינים זהה ל (5.1.1). בשלב השני מחלקים את התמונה לאזורים בגודל $2^l \times 2^l$ ברזולוציות שונות לפי $l = 0,1,2,..$ כמו ב [16]. בשלב הבא, בשונה מ [16], יוצרים היסטוגרמות של מילים באמצעות BoF מכל אחד מ 21 התאים לפי (5.1.2). לאחר מכן, סופרים את מספר המילים בכל תא (איור 9) ומחברים אותם לפי משקלים כמו ב [16]. בהמשך, משתמשים בגרעין זיווג היסטוגרמות מרחבי כדי למזג חיתוכי היסטוגרמות בצורה היררכית לכל הרזולוציות. לבסוף, מקבלים ייצוג של התמונה בצורת היסטוגרמה המכילה את איחוד כל הרזולוציות.

יש לשים לב לכך שברזולוציה הנמוכה ביותר $l = 0$, התא בגודל 1×1 מייצג למעשה את כל התמונה. חלוקת המאפיינים שנמצאים בו לתדירויות זה בדיוק מקרה פרטי של BoF.



איור 9 – בניית היסטוגרמה ממאפיינים מרחביים מממדים שונים מהעבודה [7]

בדוגמה זו רואים חלוקה של התמונה לפירמידה בעלת שלושה שלבים. לתמונה יש שלושה סוגי מאפיינים: עיגול, יהלום וסימן חיבור. בשורה העליונה ניתן לראות חלוקה של התמונה לשלושה שלבי רזולוציה שונים. לאחר מכן, לכל שלב ולכל סוג מאפיין סופרים את המאפיינים שנמצאים בכל תא ברזולוציה הרלבנטית. בשלב האחרון נותנים משקל לכל דיאגרמה מרחבית. משקל זה מציין את חוזק ההתאמה בין ההיסטוגרמות. כלומר, מאפיינים שנמצאו באותו התא ברזולוציה גבוהה יקבלו משקל יותר גבוהה מאשר כאלו שנמצאו באותו התא ברזולוציה נמוכה.

6 קידוד דליל

ישנם לא מעט מאמרים ועבודות בנושא קידוד דליל Sparse Coding (SC) [33] [34] [35]. נפרט פה את המודל הבסיסי המשותף של הבעיה.

תיאור פורמלי:

יהיה $y \in \mathbb{R}^n$ נקודה בסט וקטורים Y. ניתן להציג את y באופן הבא:

$$y \approx Da$$

כאשר:

- $D \in \mathbb{R}^{n \times m}$ הוא אוצר מילים המיוצג על ידי מטריצה.
- $a \in \mathbb{R}^m$ הוא וקטור שרוב אבריו הם אפסים. בנוסף, וקטור a הוא ייצוג דליל של y.
- לרוב באוצר המילים D מספר השורות נמוך ממספר העמודות כלומר, $(n < m)$. צורה זו נקראת *over-complete*. העמודות של אוצר המילים נקראות 'אטומים'.

המטרה:

למצוא התאמה $d(y)$ ואוצר מילים D כך ש:

$$\hat{y} = \mathrm{Dd(y)}$$

כאשר $\hat{y}$ צריך להיות מקורב עד כמה שניתן ל y.

הלמידה באופן כללי:

יצירת אוצר מילים D כך שלכל $y \in Y$ יהיה ייצוג דליל.

הבדיקה באופן כללי:

נניח כי y הוא ייצוג מקומי של אזור בתמונה, ו D הוא אוצר מילים נתון. צריך למצוא את ווקטור a שמהווה ייצוג דליל של y.

בעיית הקידוד הדליל ניתנת לפתרון מקורב בצורה טובה בעזרת אופטימיזציה.

ייצוג בעיית האופטימיזציה:

$$\min_{D,d(y)} \sum_{y} \|\hat{y} - Dd(y)\|^2 + \lambda S(d(y))$$

כאשר:

- החלק $S(d(y))$ מווסת דלילות.
- החלק $\|\hat{y} - Dd(y)\|^2$ מבצע את ההתאמה.
- הפרמטר λ נקבע על ידי המשתמש לווסת דלילות.

- ההתאמה S יכולה להיות:
  - נורמה l_1 – כפי שמתואר ב [11].
  - נורמה l_2
  - $\|d_i \odot c_i\|$ – כפי שמתואר ב [12].

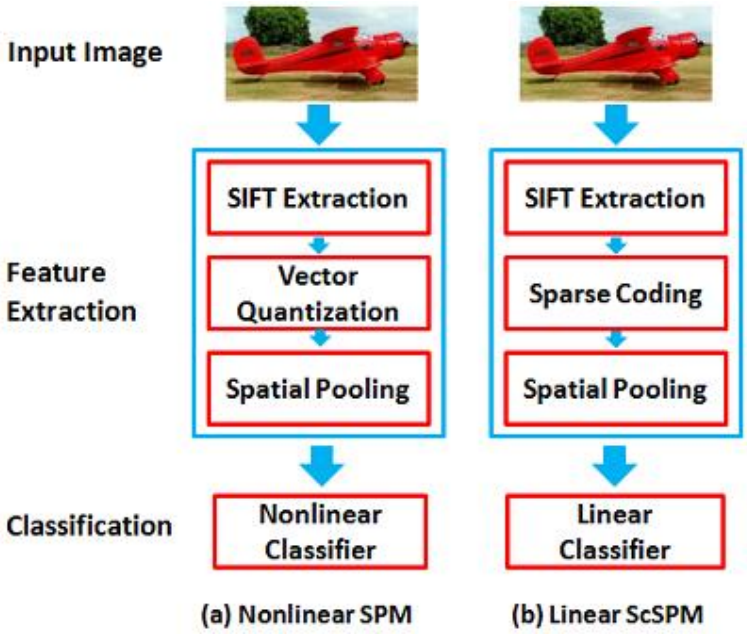
בדרך כלל פותרים את בעיית האופטימיזציה הזו בעזרת אלגוריתמים חמדנים כמו Gradient Descent. בעבודה [12] שאסקור בהמשך השתמשו ב Coordinate Descent שזו שיטה מאוד דומה ל Gradient Descent בה לא מחשבים שיפועים, אלה מתקדמים לאורך הצירים.

7 זיווג לינארי דמוי פירמידה מרחבי על בסיס קידוד דליל

השיטה Sparse coded SPM (ScSPM) [11] היא הרחבה ל SPM על ידי שינוי שלב הפקת המאפיינים ושימוש במסווג SVM לינארי. כפי שראינו בסעיף (5.2), מסווג ה SVM בו השיטה SPM משתמשת הוא לא לינארי, מה שנותן סיבוכיות $O(n^2 \sim n^3)$ בשלב הלימוד ו $O(n)$ בשלב הבדיקה כאשר n הוא גודל סט האימון.

ההרחבה בשיטה ScSPM נותנת סיבוכיות של $O(n)$ בשלב הלימוד וסיבוכיות קבועה בשלב הבדיקה מבלי לפגוע בדיוק הסיווג. למעשה, במספר מבחני ביצועים שעשו ב [11] התוצאות שקיבלו הראו שדיוק הסיווג של SPM לינארי עם מתארי SIFT בעלי קידוד דליל טובות יותר מאשר SPM לינארי על היסטוגרמות של מאפיינים.

התוצאות הללו התקבלו כאשר בזמן הפקת המאפיינים מהתמונה החליפו את שלב קידוד מתארי ה SIFT בשלב בו מבצעים קידוד דליל. את הקידוד מבצעים על מתארי SIFT ברזולוציות שונות במרחב התמונה עם גרעין SPM לינארי ובשלב הסיווג משתמשים במסווג SVM לינארי. (איור 10) מראה בצורה סכמתית את ההבדלים בין השיטות SPM ו ScSPM.



איור 10 – השוואה בין SPM מהעבודה [7] לא לינארי לבין השיטה ScSPM מהעבודה [11]

פונקציית הצבירה של המאפיינים במרחב ב SPM לא לינארי היא מיצוע (K-means) בעוד פונקציית הצבירה של המאפיינים במרחב ב ScSPM היא צבירה מקסימלית (max pooling).

7.1 רקע

העבודה מציעה גישה לייצוג תמונה על ידי פירמידה מרחבית הבנויה מקודים דלילים (וקטורים שברובם מכילים אפסים) של מתארי SIFT. כלומר, במקום לבצע קידוד מתארי SIFT באמצעות K-means, מקודדים את המתארים באמצעות קידוד דליל (Sparse Coding - SC). מכיוון שהשיטה הזו לא משתמשת בקידוד היסטוגרמות היא יותר חסינה מקומית להעתקות הזזה – זאת מכיוון ש SC שומר על התכונות של מתארי ה SIFT המקוריים מקומית בתמונה. בנוסף, מבדיקות עולה שייצוג תמונה בשיטה זו מורכב מיותר מאפיינים השייכים לתבנית הכללית של האובייקט המצולם. הבדיקות עוד מעלות כי הייצוג של המידע הוויזואלי בשיטה זו מתאים בצורה טובה מאוד למסווגים ליניאריים.

7.2 המעבר מקידוד מתארים לקידוד דליל

אחרי שלב (5.1.1) יש בידינו סט X של וקטורי SIFT מממד D. נסמן אותם כך:

$$X = [x_1, x_2, \dots, x_M]^T \in \mathbb{R}^{M \times D}$$

בשלב 14 מבצעים VQ על ידי הרצת K-means כדי לחלץ אוצר מילים מהתמונה כך:

(1.7) – אופטימיזציה K ממוצעים

$$\min_{V} \sum_{m=1}^{M} \min_{k=1\ldots K} \|x_m - v_k\|^2$$

הקבוצה $V = [v_1, \ldots, v_K]^T$ היא קבוצת K הווקטורים אותם רוצים למצוא. הביטוי $\|\cdot\|$ מציין נורמה L_2 של הווקטורים. כלומר, המרחק האוקלידי ביניהם. ניתן לנסח את בעיית האופטימיזציה המתוארת ב (1.7) באופן הבא:

(2.7) –אופטימיזציה קידוד דליל ScSPM חלק ההתאמה

$$\min_{V,U} \sum_{m=1}^{M} \|x_m - u_m V\|^2$$

על (2.7) חלים האילוצים הבאים:

(3.7) – אילוץ עוצמה

$Card(u_m) = 1$ רק איבר אחד מ u_m שונה מ 0, $\forall m$.

(4.7)- אילוץ נורמה L1

$|u_m| = 1$ סכום בערך מוחלט של כל אברי u_m הוא 1, $\forall m$.

(5.7) – אילוץ סימן

$u_m \geqslant 0$ כל אברי u_m חיוביים, $\forall m$.

בצורה זו, אחרי האופטימיזציה של V ו U, האינדקס של אברי u_m השונים מ 0 יציין את האשכול אליו שייך x_m. מכאן שבשלב ה VQ לפי (2.7) מנסים למצוא V ו U שמקיימים את קבוצות הפתרונות. בשלב הבדיקה לפי (2.7) הווקטורים בקבוצה שנלמדה V משמשים להתאמה של קבוצות חדשות של וקטורים X נתונים והמשוואה (2.7) נפתרת לפי U בלבד.

האילוץ (3.7) לרוב נותן שחזור גס של X בכדי לקבל שחזור יותר עדין, "מקלים" על האילוץ על ידי דרישה של נורמה L_1 על u_m. נורמה L_1 מאלצת מעט איברים שונים מאפס ב u_m.
בדרך זו מקבלים הגדרה חדשה לשלב VQ הנקראת Sparse Coding (SC):

(6.7) - אופטימיזציה קידוד דליל ScSPM עם ביטוי הרגולציה

$$\min_{V,U} \sum_{m=1}^{M} \|x_m - u_m V\|^2 + \lambda |u_m|$$

על (6.7) חל האילוץ:

(7.7) – אילוץ גודל

$$\|v_k\| \leq 1, \forall k = 1,2,\dots,K$$

בדרך כלל, אוצר המילים V הוא בסיס שמרחב הווקטורים שמרכיבים אותו גדול מהממד שלהם, כלומר $K > D$.

7.3 תהליך הלמידה עם קידוד דליל בקווים כלליים

הלימוד ב SC כולל את השלבים הבאים:

אימון:

- חילוץ סט X של מתארים מתמונות של אובייקטים.
- פתרון (6.7) עם סט X לפי V כאשר U קבוע ופתרון V ישמש כאוצר המילים.

כאשר U קבוע הבעיה מצומצמת לפתרון ריבועים פחותים (נספח ב):

(8.7) – ניסוח פתרון בעזרת ריבועים פחותים

$$\min_V \|X - UV\|_F^2$$

עם האילוצים:

$$\|v_k\| \leq 1, \forall k = 1,2,\dots,K$$

בדיקה:

- חילוץ סט X של מתארים מתמונה של אובייקט מסוים.
- פתרון (6.7) עם סט X לפי U, V נתון כאוצר המילים שנלמד בזמן האימון.

כאשר V משמש כאוצר המילים הבעיה ניתנת לפתרון על ידי רגרסיה לינארית על כל u_m כך:

(9.7) – ניסוח פתרון בעזרת רגרסיה

$$\min_{u_m} \|x_m - u_m V\|^2 + \lambda |u_m|_{L_1}$$

פתרון בעיית האופטימיזציה (9.7) מבוצע בעזרת אלגוריתם feature-sign search כמו שמתואר בעבודה [33].

7.4 סיווג על ידי זיווג לינארי דמוי פירמידה מרחבי

בצורה הבסיסית של BoF מייצגים את המידע הוויזואלי באמצעות סט לא סדור של תדירויות מאפיינים (היסטוגרמת מאפיינים) שקודדו בצורה כזו או אחרת. צורה נפוצה לחשב היסטוגרמת מאפיינים מאוסף שכזה נראית כך:

(10.7) – ניסוח SPM

$$z = \frac{1}{M}\sum_{m=1}^{M} u_m$$

ההיסטוגרמה z מחושבת בצורה זו לכל תמונה, כאשר U זו קבוצת הווקטורים שכל אחד מהם מייצג התאמה למילה במילון V לפי (2.7).

בשיפור SPM ל BoF ייצוג הפירמידה המרחבי של התמונה על ידי ההיסטוגרמה z מורכב משרשור היסטוגרמות מקומיות (11.7) בחלקים שונים של התמונה ברזולוציות שונות.

כדי לסווג את ההיסטוגרמה z_i המייצגת את התמונה I_i, על מסווג SVM בינארי ללמוד את פונקציית הסיווג הבאה:

(11.7) – מבנה פונקציית הסיווג

$$f(z) = \sum_{i=1}^{n} \alpha_i k(z, z_i) + b$$

סט הלימוד הוא קבוצת הזוגות $\{(z_i, y_i)\}_{i=1}^{n}$ כאשר התוויות $y_i \in \{-1, +1\}$ מוצמדות לכל מילה z_i במילון. כדי לסווג תמונה z נבדוק את אי השוויון $f(z) > 0$ אם התוצאה חיובית התמונה תסווג עם התווית החיובית ואם התוצאה שלילית התמונה תסווג עם התווית השלילית.

אחרי לא מעט עבודות וניסויים נמצא על ידי [32] כי פונקציות הגרעין שנותנות תוצאות טובות על היסטוגרמות מאפיינים הן (2.5) ו- (5.5). הגרעינים הללו הם לא לינאריים ולכן היעילות החישובית שלהם חסומה על ידי $O(n^3)$ ו- זיכרון $O(n^2)$ כאשר n מייצג את מספר המתארים.

העבודה ScSPM מציגה שיטה חדשה לייצוג תמונה המשתמשת במסווג SVM לינארי מבוסס קידוד דליל של מתארי SIFT. הסט U שמתקבל מ (6.7) בעזרת סט האימון X ואוצר המילים שנלמד V משמש לייצוג תמונה z על ידי פונקציית הצבירה $\mathcal{F}$ כך:

(12.7) – סימון פונקציית הצבירה

$$z = \mathcal{F}(U)$$

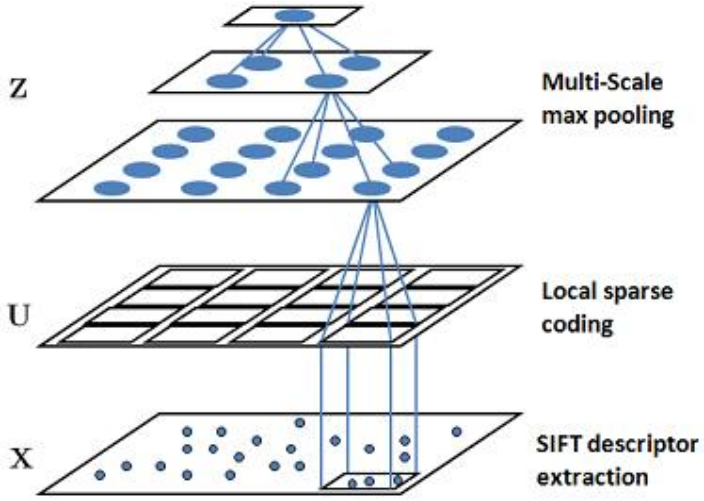
פונקציית הצבירה $\mathcal{F}$ מוגדרת על כל עמודה של U בצורה הבאה:

(13.7) – הגדרת פונקציית הצבירה

$$z_j = \max\{|u_{1j}|, |u_{2j}|, \dots, |u_{Mj}|\}$$

z_j הוא האיבר ה - j של z ו - u_{ij} הוא האיבר בשורה ה – i והעמודה ה – j של המטריצה U. מספר המאפיינים המקומיים באזור מסוים בתמונה נתון על ידי M.

מכיוון שכל עמודה ב U מייצגת קידוד דליל של קבוצת מאפיינים מקומיים המייצגת מילה במילון V, פונקציית הצבירה נותנת לנו ייצוג היררכי של קודים דלילים של מאפיינים באזורים שונים של התמונה (איור 11).



איור 11 – תיאור סכמתי של הארכיטקטורה של ScSPM מהעבודה [11]

X מייצג את קבוצת המאפיינים שנאספו מהתמונה. הווקטור U הוא הייצוג הדליל של מאפיינים אילו כאשר ייצוג זה מהווה מידת זיקה בין כל מאפיין למילה במילון. את קודים אלו צוברים בצורה היררכית מאזורים שונים בתמונה ברזולוציות שונות.

בעבודה זו מצאו כי מאפיינים שנאספו על ידי פונקציית צבירה מאזורים שונים וברזולוציות שונות של תמונה היו חסינים לטרנספורמציות מקומיות וזה שיפור משמעותי של BoF שמשתמשת בממוצע של מאפיינים שנאספים מכל התמונה ולא מחלוקה מרחבית של התמונה לאזורים.

7.5 גרעין זיווג לינארי דמוי פירמידה מרחבי

פונקציית גרעין ה SVM של ScSPM מוגדרת בצורה הבאה:

(14.7) – גרעין SPM

$$k(z_i, z_j) = z_i^T z_j = \sum_{l=0}^{2} \sum_{s=1}^{2^l} \sum_{t=1}^{2^l} \langle z_i^l(s,t), z_j^l(s,t) \rangle$$

התמונה I_i מיוצגת על ידי z_i כאשר:

- $\langle z_i, z_j \rangle = z_i^T z_j$
- $z_i^l(s,t)$ – צבירה של הקודים הדלילים באזור ה (s,t) של התמונה I_i ברזולוציה l.
לדוגמה, אם $l = 4$ התמונה מחולקת ל $2^4 = 16$ אזורים. כאשר t ו s הם כל הזוגות (s,t) כך ש $s,t \in \{1,2,\dots l\}$. למשל אזורים $(1,4), (2,1)$ וכדי.

פונקציית הלמידה של SVM מוגדרת, אם כן, בצורה הבאה:

(15.7) – הגדרת פונקציית הלמידה SVM

$$f(x) = \left(\sum_{n=1}^{n} a_i z_i \right)^T z + b = w^T z + b$$

כפי שניתן לראות, היעילות החישובית של הגרעין תלויה במספר המאפיינים n ובמספר הרזולוציות שהוא קבוע 2^l. ולכן, היעילות של הגרעין היא $O(n)$ ויעילות הלמידה של $f(x)$ היא לינארית. יתר על כן, היעילות החישובית של הסיווג לכל תמונה היא קבועה.

8 סיווג תמונות בעזרת קידוד לינארי מקומי

השיטה Locality-constrained Linear Coding for Image Classification (LLC) [12], מציעה גם היא שיפור ל SPM המבוסס על BoF בשלב קידוד מתארי ה SIFT (Vector Quantization) של השיטה. בדומה ל ScSPM, במקום VQ של מתארי SIFT, LLC מקודדת ומשתמשת במיקום של כל מתאר SIFT כדי להטיל אותו למערכת קואורדינטות לפי מיקומו במרחב. לאחר מכן, כל הקואורדינטות המוטלות משולבות באמצעות Max Pooling כדי ליצור את הייצוג של התמונה.

ישנם כמה חידושים בשיטה LLC. בשלב הראשון אוצר מילים נבנה על ידי אלגוריתם K השכנים הקרובים ביותר (נספח ג) על מתארי ה SIFT שנאספים מהתמונה. בשלב הבא מקודדים את המילים באוצר המילים על יד שיטת הריבועים הפחותים (נספח ב). המידע המקודד מסווג באמצעות SVM לינארי. עבודה [36] וגם עבודה זו הן שיפור של ScSPM מבחינת הצורך שהמאפיינים המקודדים יהיו מקומיים לאזורים בתמונה ובכך משיגים דיוק מרבי בסיווג. עבודה זו משיגה שיפורים נוספים לעומת ScSPM ו [36]. שיפור אחד הוא שלא נדרש באף שלב לפתור בעיית אופטימיזציה של נורמה מסדר L_1. שיפור שני הוא שפונקציית המטרה מחושבת בצורה מתמטית והיא לא תלוית אופטימיזציה.

8.1 רקע

בעבודה [36] מראים כי ישנה חשיבות מרובה למיקום המתארים בתמונה וישנה עדיפות למתארים מקומיים על אילוצי קידוד וקטורים דלילים. הם מראים כי קידוד מתארים מקומיים בתמונה מאלץ קודים דלילים אבל לא להפך. לכן, LLC מחייב מתארים מקומיים ולא וקטורים דלילים שכן קידוד דליל יתקבל כך או כך.

8.2 בניית אוצר מילים

אוצר המילים מופק בצורה אופטימלית בכמה שלבים. בשלב הראשון, משתמשים באלגוריתם K ממוצעי אשכולות כדי לקבל את אוצר המילים הבסיסי B. לאחר מכן, מאמנים את אוצר המילים בעזרת סט X של מתארים כדי לקבל אוצר מילים אופטימלי באופן הבא:

בהינתן סט X של N מתארים מממד D. נסמן אותם כך:

$$X = [x_1, x_{2,} \dots, x_N] \quad \in \mathbb{R}^{D \times N}$$

ואוצר מילים בסיסי B עם M מילים מממד D שיסומן כך:

$$B = [b_1, b_{2,} \dots, b_M] \quad \in \mathbb{R}^{D \times M}$$

בניית אוצר המילים לפי LLC מנוסחת באופן הבא:

(1.8) – ניסוח אופטימיזציה LLC

$$\underset{C,B}{argmin}\sum_{i=1}^{N}\|x_i - Bc_i\|^2 + \lambda\|d_i \odot c_i\|^2$$

- על (1.8) חלים האילוצים הבאים:

(2.8) – אילוץ עוצמה

$$1^T c_i = 1, \forall_i$$

(3.8) – אילוץ גודל

$$\|b_j\|^2 \leq 1, \forall_j$$

- האופרטור הבינארי $\odot$ מציין את מכפלת הקואורדינטות של הווקטורים בהתאמה, לדוגמה: $(1,2)\odot(3,4) = (3,8)$.
- האבר $d_i \in \mathbb{R}^M$ משמש כמדד להתאמה מקומית בתמונה ל x_i.

d_i מוגדר באופן הבא:

(4.8) – ניסוח התאמה LLC

$$d_i = \mathrm{e}^{\left(\frac{dist(x_i,B)}{\sigma}\right)}$$

ככל ש d_i קטן, כך הוא מתקרב לערכים של הבסיס וההתאמה בינו לבין c_i גבוהה יותר.

- הפונקציה $dist(x_i, B)$ מוגדרת באופן הבא:

$$dist(x_i, B) = [dist(x_i, b_1), \dots, dist(x_i, b_M)]^T$$

- $dist(x_i, b_j)$ הוא המרחק האוקלידי בין b_i ל x_i.
- הקבוע σ משמש למיתון ההתאמה של d_i. הוא נקבע בצורה יוריסטית כפי שנראה בהמשך.

8.3 קידוד מתארים

הקידוד המקומי של המתארים מתבצע בעזרת אלגוריתם K ($K < D < M$) השכנים הקרובים ביותר היררכי לכל x_i . השכנים הללו מהווים את אוצר המילים המקומי B_i עבור x_i. הקידוד של x_i יתקבל באופן הבא:

(5.8) – ניסוח אופטימיזציה לקידוד LLC

$$\underset{\tilde{C}}{min}\sum_{i=1}^{N}\|x_i - \tilde{c}_i B_i\|^2$$

- על (5.8) חל האילוץ הבא:

$$1^T \tilde{c}_i = 1, \forall_i$$

ייצוג התמונה בשיטה זו מבוצע בדיוק כמו ב ScSPM על ידי פונקציית צבירה של קידוד המתארים מעל חלוקה של התמונה ל L אזורים ברזולוציות שונות.

8.4 יתרונות השיטה

- **דיוק בשלב הבדיקה**. ב BoF קידוד המתארים למילים במילון נעשה כפי שמתואר ב (5.1.3) על ידי פעולת VQ. פעולה זו מייצרת שגיאה שצריך לתקנה בשלב הבדיקה. השגיאה טמונה באלגוריתמי חלוקה לאשכולות אשר עלולים לסווג מספר המתארים שונים של התמונה לקידוד מסוים במילון מחד. ומאידך, מתארים של אותה המילה עלולים להיות מסווגים למילים שונות במילון. מסיבה זו נדרשים להשתמש בגרעין לא ליניארי בשלב הבדיקה כדי לפצות על שגיאות הקידוד.

  לעומת זאת ב LLC, מתארים מיוצגים על ידי אוסף של מילים (max pooling). יתר על כן, בזכות החלק המתקן $\|d_i \odot c_i\|^2$, LLC מאפשרת ייצוג מורכב של מאפיינים שמאבד פחות מידע. ייצוג זה אף מבטא את מידת ההתאמה בין המילים השונות במילון. בזכות צורה זו של ייצוג יותר קל לשחזר את המידע בצורה מדויקת ובעזרת גרעין לינארי. הדיוק נובע מהסיבה שמאבדים פחות מידע בשלב הקידוד ובנוסף הקידוד מקבל משקל יותר גבוהה ככל שהמילים יותר "קרובות" אחת לשנייה.

- **התאמה מקומית דלילה**. ההבדל העיקרי בין ScSPM לבין LLC, הוא הביטוי המתקן $\|d_i \odot c_i\|^2$ הנותן משקל רב לקרבה של מתארים מקומיים מאשר לדלילות הקידוד ולכן חלקי תמונה זהים ייוצגו על ידי מילים זהות במילון. ב ScSPM הביטוי המתקן $|u_m|$ (נורמה L_1) מאלץ קידוד דליל וכתוצאה מכך תכונות זהות של התמונה לעיתים מיוצגות על ידי צרוף של מילים שונות - "רחוקות" אחת מהשנייה.

- **חישוב מתמטי**. ב LLC לא נידרש לבצע אופטימיזציה על הביטוי המתקן $\|d_i \odot c_i\|^2$. הוא מחושב בצורה מטריציונית על ידי מטריצת השונות המשותפת (covariance matrix) באופן הבא:

(6.8) – מטריצת השונות המשותפת המוגדרת על ידי C_i

$$C_i = (B - 1x_i^T)(B - 1x_i^T)^T$$

(7.8) – נוסחה לחישוב מתמטי של הקודים ב LLC

$$\tilde{c} = (C_i + diag(d))\backslash 1$$

9 תוצאות

בפרק זה אנתח את ההישגים של שיטות הסיווג מפרקים 7 ו 8 ואערוך השוואה בין התוצאות שאליהן הגיעו.

השיטות נבדקו על מספר מאגרי נתונים של תמונות שהפכו לבסיס עליו מסתמכות לא מעט עבודות מודרניות העוסקות בראייה ממוחשבת. בעזרת מאגרי תמונות אלו והמגוון הרחב של התמונות בהן ניתן לבדוק את הביצועים של השיטות השונות ולהשוות אותם בצורה תקנית לביצועי שיטות אחרות כמו שנראה בהמשך.

העבודות שסקרתי בעיקר נבדקו על מאגרי התמונות Caltech-101 - [8] ו Caltech-256 - [10]. לכל מאגר תמונות יש יחוד משלו כפי שאסביר להלן.

9.1 מאגר התמונות Caltech-101

מאגר תמונות זה מכיל מעל 9,000 תמונות מסווגות ל 101 מחלקות שונות ביניהן מחלקות של מטוסים, רכבים, פרחים, פנים וכד׳ ועוד מחלקה אחת לרקע של תמונות. בכל מחלקה של תמונות ישנן בין 31 ל 800 תמונות. בכל בדיקה נבחר מספר תמונות שונה לסט האימון של מחלקה מסוימת, בדרך כלל 5, 10, ... , 50 תמונות לסט. אני אתייחס לתוצאות שהושגו מסטים של 15 ו 30 תמונות מכוון שזה מספר התמונות הנהוג בבדיקות של מאגר תמונות זה. בכל הבדיקות SPM חולק ל 3 רזולוציות של התמונה: $1 \times 1,\ 2 \times 2,\ 4 \times 4$.

תמונות האימון הותאמו לגודל 300×300 פיקסלים כאשר נשמר היחס בין הרוחב לגובה בתמונה.

התוצאות שמוצגות ב (טבלה 1.9) הן הממוצע של הדיוק בסיווג בין 101 המחלקות ומחלקת תמונות הרקע.

9.1.1 תוצאות LLC על Caltech-101

בבדיקות של LLC הרכיבו את אוצר המילים מ 2048 מילים.

התוצאות הראו 100% דיוק בזיהוי כאשר השתמשו בסט אימון של 30 תמונות של 13 מחלקות ביניהן מחלקת הרכבים, הכיסאות, הנמרים ועוד.

9.1.2 תוצאות ScSPM על Caltech-101

בבדיקות של ScSPM הרכיבו את אוצר המילים מ 1024 מילים.

בכל הבדיקות בהשוואה לשיטות SPM אחרות כמו [7] [16] ScSPM הראתה ביצועים ודיוק רב יותר.

30	15	שיטות / # תמונות אימון
66.2	59.1	Zhang [37] – SVM-KNN
70.4	65.00	Boiman [38] – NBNN
69.10	61.00	Jain [39] – ML– CORR
64.16	-	Gemert [40] – KC
64.60	56.40	Lazebnik [7] – KSPM
58.81	53.23	Grauman [16] – LSPM
73.20	**67.00**	Yang [11] – ScSPM
73.44	65.43	Wang [12] – LLC

טבלה 1.9 – תוצאות באחוזי דיוק על מאגר התמונות Caltech-101

9.2 מאגר התמונות Caltech-256

מאגר תמונות נרחב זה מכיל מעל ל 30,000 תמונות המסווגות ל 265 מחלקות שונות של אובייקטים. מאגר תמונות זה הוא שיפור של מאגר 101 בכמה אופנים. הוא מכיל מגוון רחב יותר של גדלים של אובייקטים, מגוון מיקומים בתמונה, מגוון רחב של תנוחות ושונות תוך מחלקתית. בכל מחלקה של תמונות ישנן לפחות 80 תמונות. נוספה עוד מחלקה של תמונות רועשות (clutter) לאימון וייצוג יותר טוב של רקע של תמונות.

כמו בבדיקת המאגר Caltech-101, בכל בדיקה נבחר מספר תמונות שונה לסט האימון של מחלקה מסוימת. מספר התמונות הנהוג בבדיקות של מאגר תמונות זה הוא: 15, 30, 45, 60 תמונות לסט.

בכל הבדיקות SPM חולק ל 3 רזולוציות של התמונה: $4 \times 4,\ 2 \times 2,\ 1 \times 1$.

תמונות האימון הותאמו לגודל 300×300 פיקסלים כאשר נשמר היחס בין הרוחב לגובה בתמונה. התוצאות שמוצגות הן הממוצע של הדיוק בסיווג בין 256 המחלקות השונות.

ראוי לציין שהשיטה LLC הראתה תוצאות טובות יותר בכל הבדיקות. בנוסף, בבדיקה זו השתמשו ב 2096 מילים באוצר המילים שנלמד וזמן הבדיקה הממוצע לעיבוד כל תמונה הוא 0.3 שניות שהוא זמן מרשים ביותר בהתחשב בעובדה שהבדיקות בוצעו על מכונה מסוג "Dell PowerEdge 1950 server with 16G memory and 2.5Ghz Quad Core CPU". זו אמנם מכונה חזקה יחסית, אולם ישנן מכונות בעלות חומרה חזקה שיכולות לתת ביצועים טובים אף יותר.

השיטה LLC מראה דיוק של מעל ל 90% בסיווג 20 מחלקות מתוך מאגר התמונות ביניהן: פרחים, פנים, מטוסים, בניינים ועוד.

השוואת תוצאות בין השיטות השונות על Caltech-256 מוצגת ב (טבלה 2.9). התוצאות של אלגוריתמים KC [40] ו KSPM [7] כאשר סט הלימוד גדול מ 30 תמונות לא נבחנו מכיוון שזמן הלימוד וצריכת המשאבים בהן גדול מדי. עובדה זו מראה יתרונות מובהקים של ScSPM ו LLC שבהם דנו בפרקים 6 ו 8 על שיטות אחרות בכך שהן יותר מהירות, צורכות פחות משאבים והדיוק שלהן עדיין גבוה יחסית.

שיטות / # תמונות אימון	15	30	45	60
Gemert [40] – KC	-	27.17	-	-
Lazebnik [7] – KSPM	-	34.10	-	-
Grauman [16] – LSPM	13.20	15.45	16.37	16.57
Yang [11] – ScSPM	27.73	34.02	37.46	40.14
Wang [12] – LLC	**34.36**	**41.19**	**45.31**	**47.68**

טבלה 2.9 – תוצאות באחוזי דיוק על מאגר התמונות Caltech-256

כפי שניתן ללמוד מהתוצאות, השיטה ScSPM מראה דיוק רב על מאגר התמונות Caltech 101 לעומת LLC כאשר מספר דוגמאות האימון קטן – 15 ; ותוצאות מאוד קרובות ל LLC כאשר מספר דוגמאות האימון עולה – 30. ניתן לייחס לתוצאות אלו את העובדה שמאגר התמונות מהווה כמעט את אותו האתגר עבור שתי השיטות במובן שבמאגר זה ישנו פחות גיוון במיקום האובייקטים והתנוחות השונות שלהם. לעומת זאת, התוצאות על מאגר התמונות Caltech 256 מראות עליונות ניכרת של LLC לעומת תוצאות שאר השיטות ובפרט מול תוצאות ScSPM. באופן כללי נראה כי ככל שיש יותר דוגמאות בסט האימון של ScSPM ו LLC הדיוק משתפר עד גבול מסוים של התכנסות.

ראוי לציין שישנם עוד מאגרים ואתגרים שנבחנו בעבודות שסקרתי כמו: Pascal VOC 2007 - [9], 15 Scenes categorization - [41] [7] ו- TRECVID 2008 Surveillance Video - [42], אולם לא ניתן להשוות בין התוצאות של LLC ו ScSPM עליהם שכן כל עבודה בחרה להציג את תוצאותיה על אתגר או מאגר תמונות אחר.

9.3 דיון

הדיון מחולק למספר תתי סעיפים כאשר בכל סעיף אנתח מרכיב אחר בשיטות שסקרתי ואיך הוא משפיע על התוצאות שהוצגו.

9.3.1 מתארים

כמו שהסברתי בסעיף (5.1.1) כדי שהמתארים יהיו חסינים לשינוי קנה מידה, אחד מהשלבים של ייצוג SIFT הוא לשמור מידע על מתארים מקומיים בתמונה בכמה קני מידה.

בעבודה LLC השתמשו במתארים מסוג Histogram of Oriented Gradient (HoG) [24] שנאספו מחלקי תמונה מקומיים בקני מידה 31×31, 25×25, 16×16. לעומת זאת, תוצאה מאוד מעניינת של ScSPM היא שהשתמשו במתארי SIFT רק מקנה מידה 16×16. ב ScSPM מדווחים שכאשר בוצעו בדיקות עם מאפיינים שנאספו מקני מידה שונים הם לא ראו שיפור משמעותי בתוצאות. תוצאה זו, כמו שמדווח ב [11] סעיף (5.5.1), מעידה כנראה על כך שפונקציית הצבירה על קודים דלילים מצליחה להתגבר על הצורך לשמור תכונות מקומיות של אובייקטים בקני מידה

שונים ולכן לא נדרש להשתמש בטכניקות של חסינות לקנה מידה. להערכתי זה נכון עד גבול מסוים מכוון שאמנם העבודה ScSPM הראתה תוצאות טובות יותר משאר העבודות אשר בחלקן השתמשו במאפייני SIFT בקני מידה שונים על מאגר התמונות Caltech-101 (טבלה 1.9) אבל כאשר היא נבדקה על מאגר התמונות Cltech-256 שבו יש יותר מגוון של קני מידה של אובייקטים, התוצאות אמנם היו טובות יותר מאשר התוצאות של שאר העבודות (טבלה 2.9) אבל לא יותר טובות משל LLC בה המתארים נדגמו בכמה קני מידה כמו שמפורט לעיל.

9.3.2 אוצר המילים

ב ScSPM אוצר המילים נבנה לפי התהליך והאלגוריתמים המתוארים בסעיף (7.3).
גודלו של אוצר המילים הוא בעל חשיבות מכרעת המשפיעה על הדיוק של השיטה בצורה משמעותית. ניתן לראות לפי (טבלה 3.9) שהדיוק של השיטה על מאגר התמונות Caltech-101 גדל ככל שהשתמשו ביותר קודים, כאשר השיטה SPM המקורית [16] מגיעה להתכנסות סביב 512 קודים. בבדיקות שערכתי הגדלתי את מספר הקודים ל 2048 ומצאתי כי זהו מספר הקודים סביבו השיטה מתכנסת ומגיעה לדיוק הרב ביותר. את האינטואיציה לערוך ניסוי זה עם הקוד שפורסם על ידי ScSPM קיבלתי כאשר בחנתי את תוצאות LLC על מאגר התמונות Caltech-101. על אוצר מילים זה גודל המילון שהראה תוצאות אופטימליות היה 2048.

שיטות / גודל אוצר המילים	256	512	1024	2048
Grauman [16] - LSPM	61.97	63.23	69.70	-
Yang [11] - ScSPM	**68.26**	**71.20**	**73.20**	**73.91**
Wang [12] - LLC	-	-	72	73.44

טבלה 3.9 – השפעת גודל אוצר המילים על הביצועים

באופן כללי ניתן להסיק שככל שהמילון קטן יכולת האבחנה של האלגוריתם קטנה, וככל שישנם יותר קודים במילון מאפיינים דומים לא יסווגו לאותה המחלקה.

9.3.3 ביצועים

ב LLC אוצר המילים נבנה לפי התהליך המתואר בסעיף (8.2). ניתן לראות לפי (טבלה 3.9) שהדיוק של השיטות LLC ו ScSPM הוא גבוה יחסית והתוצאות של השיטות די קרובות אחת לשנייה. אף על פי כן, כאשר בוחנים את הביצועים של שתי השיטות ניתן לראות יתרון ניכר ל LLC על פני ScSPM בזמנים של עיבוד התמונות, חילוץ המאפיינים, הקידוד שלהם וייצוג התמונה על ידי פונקציית הצבירה. לפי ScSPM הזמן שלוקח משלב העיבוד ועד קידוד וייצוג התמונה ממאגר התמונות Caltech-101 הוא 1 שנייה בממוצע. הזמן הזה הוא מהיר מאוד יחסית לשיטות אחרות אולם השיטה LLC מראה ביצועים טובים אף יותר המגיעים ל 0.24 השנייה.

תוצאה זו מרשימה ביותר מכמה סיבות. אחת, היא מהירה בממוצע פי 4 מ ScSPM. שתיים, הדיוק של השיטה, כאמור, מאוד גבוה ולא נפגע. שלוש, התוצאה הזו מעידה על כך שהשיטה יכולה לטפל בכמויות גדולות של תמונות וכך להוות כלי מצוין לסיווג בסביבת האינטרנט, הווידאו ועוד סביבות בהן מאגר התמונות הוא גדול מאוד.

9.3.4 הפרמטרים λ, K

הפרמטר λ

כאמור, הפרמטר החופשי λ משמש כמדד לדלילות הווקטור המקודד. ככל ש λ גדל, הקידוד יהיה יותר דליל. וזאת מכוון שהבעיות הן בעיות אופטימיזציה של מציאת מינימום. כלומר, ככל ש λ גדל הווקטור המקודד שבו הוא מוכפל (לפי שתי השיטות LLC ו ScSPM הקידוד של מאפיין x_i) יצטרך להיות יותר קטן וזאת על ידי יותר אברים עם ערך אפס. כמובן שצריך למצוא ערך אופטימלי ל λ וזאת על ידי ניסוי וטעייה.

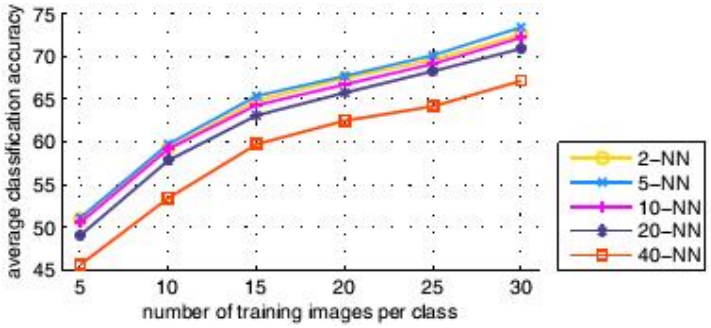
בעבודה LLC נבחר הערך $\lambda = 500$. מבדיקות שערכתי עם מספר ערכים של λ לפי הקוד שפורסם על ידי LLC נראה כי זהו באמת ערך שנותן את הדיוק המרבי על מאגר התמונות Caltech-101. ראה סעיף (10)

בעבודה ScSPM – נמצא שכאשר הם דואגים שהדלילות של הקודים היא 10% הם קיבלו תוצאות טובות. לכן, נבחרו ערכים $\lambda = 0.3 \sim 0.4$.

הפרמטר *K*

הפרמטר החופשי K משמש לקביעת מספר השכנים של אלגוריתם הסיווג K השכנים הקרובים ביותר בו השתמשו ב LLC. מבדיקות שערכו בין הערכים 2, 5, 10, 20 ו 40 נראה כי ככל שמספר השכנים קטן כך הדיוק של האלגוריתם גדל עד לגבול תחתון של שני שכנים אז התוצאות מראות שגיאה גדולה. זו תוצאה מבורכת מכוון שכך מספר החישובים קטן וצריכת הזיכרון בהתאמה קטנה. לכן נבחר הערך $K = 5$. לכאורה הערך $K = 5$ שנותן תוצאות אופטימליות באופן אינטואיטיבי נשמע כערך נמוך מדי אבל חשוב לזכור כי מציאת השכנים הקרובים ביותר באלגוריתם זה היא מקומית ומבוצעת על חלקים קטנים יותר של התמונה. כלומר, ערכים גדולים של K נותנים תוצאות לא טובות שכן הם מכילים שכנים מחלקים רחוקים יותר בתמונה. באיורים הבאים ניתן לראות הרצה שביצעתי עם *LLC* על *Caltech-101* עם מחלקה נוספת ומספר שכנים משתנה (איור 12) והתוצאות שפורסמו ב [12] (איור 13).

איור 12 – תוצאות הרצת אלגוריתם *LLC* על מאגר *Caltech-101* עם 30 תמונות אימון ומספר שכנים משתנה. למאגר התמונות בניסוי הוספתי מחלקה של תמונות נשים בהריון



איור 13 – תוצאות הרצת אלגוריתם *LLC* עם מספר שכנים ותמונות אימון משתנים כפי שפורסמו ב [12]

9.3.5 גרעין לינארי ולא לינארי

בעבודה ScSPM ביצעו השוואה של פונקציית הגרעין הלינארית שמוצעת ופונקציות הגרעין הלא לינאריות כמו (2.5) ו (5.5) שהוצעו בעבודות קודמות [16] [7].

התוצאות מראות לא רק שיפור בזמן הריצה והזיכרון אלה גם בדיוק הסיווג.

ב (טבלה 4.9) להלן ניתן לראות את התוצאות.

גרעין ScSPM	גרעין $\chi 2$	גרעין חיתוך היסטוגרמות
67.0	60.7	60.4

טבלה 4.9 – השוואה בין ביצועים של פונקציות גרעין

עדיין לא ידוע בצורה אמפירית מדוע יש התאמה בין מאפיינים המקודדים על ידי קודים דלילים לפונקציות לינאריות אך ישנן כמה השערות רווחות. אחת מההשערות היא שתבניות של מאפיינים דלילים ניתנות להפרדה לינארית ביתר קלות.

9.3.6 פונקציות צבירה

מספר פונקציות צבירה נוסו כדי להגיע לתוצאות הטובות ביותר בדיוק. כידוע פונקציות הצבירה הן ליניאריות ולכן נותנות בדרך כלל ביצועים טובים לאלגוריתמים השונים.
פונקציות הצבירה שנבדקו הן:

- צבירת מקסימום (max pooling) – כפי שמתואר בסעיף (7.5).
- צבירת שורש ריבועי של ממוצעים (Sqrt) המוגדרת כך:

$$z_i = \sqrt{\frac{1}{M}\sum_{i=1}^{M} u_{ij}^2}$$

- צבירת ממוצע ערכים מוחלטים (Abs) המוגדרת כך:

$$z_i = \frac{1}{M}\sum_{i=1}^{M} |u_{ij}|$$

התוצאות הטובות ביותר נתקבלו על ידי פונקציית צבירת מקסימום מכיוון שפונקציית צבירה זו חסינה במידה רבה יותר לשינויים מרחביים מקומיים בתמונה. בניגוד לפונקציות הצבירה השונות שנבדקו, פונקציית צבירת מקסימום משמרת את המאפיינים ולא מבצעת עליהם עיבוד מלבד נרמול l_2 שרצוי ליעילות החישובית של פונקציית הגרעין. ניתן לראות את התוצאות שנתקבלו ב (טבלה 5.9) להלן.

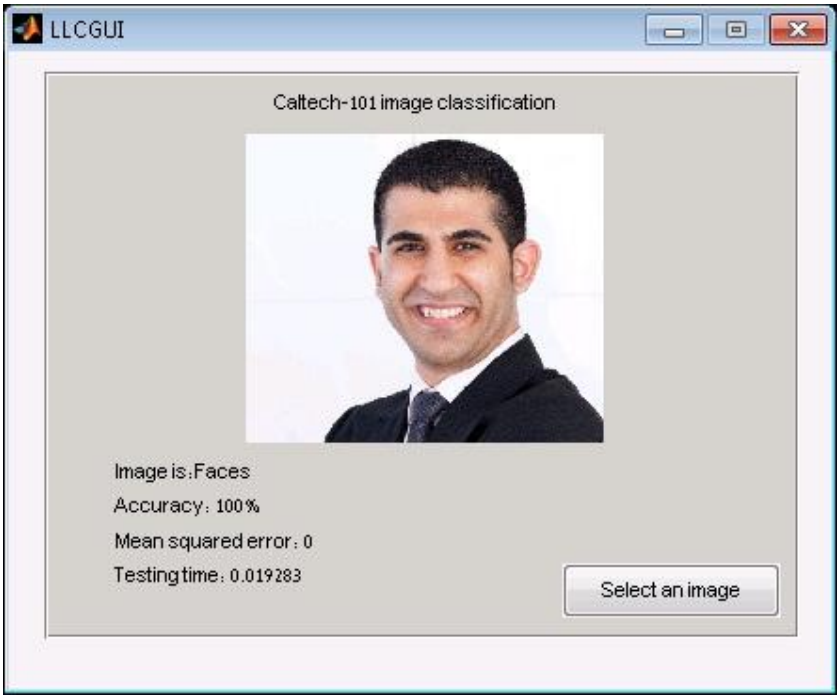
פוני צבירה / # תמונות אימון	15	30
Sqrt	-	71.09
Abs	-	66.86
Max ScSPM	**67.0**	73.2
Max LLC	65.43	**73.44**

טבלה 5.9 – בדיקת דיוק של פונקציות צבירה שונות

10 יישום LLC לסיווג אובייקטים ממאגר Caltech-101

בנוסף לסקירת השיטות, כתבתי פרויקט מעשי שמממש סיווג של אובייקטים שנלמדו מתוך מאגר התמונות Cltech-101. הפרויקט כתוב ב MATLAB ומשתמש בקוד שפורסם על ידי [12]. בנוסף, התשתית עליה מתבסס הקוד לאימון המאפיינים ובדיקת המודל היא הספרייה Liblinear [43]. ספרייה זו מממשת מסווגים לינאריים שמציגים תוצאות טובות מאוד של זמן ריצת המסווג ועבודה עם כמות של מיליוני מאפיינים. לטענתם, במערכות התומכות במרחב זיכרון של 64 סיביות, המסווגים מסוגלים לטפל בכמות של עד כ 2^{6^4} מאפיינים, כל עוד הזיכרון מתיר. הספרייה מממשת מגוון רחב של מסווגים התומכים בשיטות הסדר כמו l_1 ו l_2 על צורותיהן השונות.

הפרויקט מכיל תכנית שמציגה חלון בו המשתמש בוחר תמונה ממאגר התמונות Caltech-101 או ממקום כלשהו על המחשב עליו התוכנית רצה. אם התמונה היא אחד מ 101 האובייקטים שנלמדו מבעוד מועד, התוכנית תציג את שם המחלקה או שם האובייקט, את הדיוק, את השגיאה, ואת הזמן שלקח לסווג את התמונה. לדוגמה (איור 14) ו (איור 15).



איור 14 – דוגמת הרצה של הפרויקט כאשר נבחרה תמונה שמסווגת למחלקת פנים

```
[24-Dec-2012 01:05:06] LLC image representation - start
Processing C:\Users\enshem\Desktop\0191.jpg: wid 300, hgt 259, grid size: 48 x 41,
1968 patches
[24-Dec-2012 01:05:08] LLC image representation - end
[24-Dec-2012 01:05:08] LLC image evaluation - start
Accuracy = 100% (1/1)
[24-Dec-2012 01:05:08] LLC image evaluation - end
```

איור 15 – דוגמה לפלט של הפרויקט

בדוגמה זו ניתן לראות שהתמונה כווצה לגודל 300 x 259, המאפיינים נאספו בצורה דחוסה מטבלה בגודל 48 x 41 ומהתמונה הופקו 1968 מאפיינים

10.1 האימון והסיווג בקוד

הפרויקט בנוי משני שלבים עיקריים. שלב ה*אימון* (לא מקוון), בו עוברים על כל התמונות ממאגר התמונות Caltech-101 ויוצרים מהם מאפיינים. לאחר מכן, בונים את אוצר המילים בגודל 1024 בעזרת הרצה של K ממוצעים (נספח ד). לאחר מכן, בונים ייצוג LLC לכל תמונה בעזרת פירמידת מאפיינים ברזולוציות [1,2,4] וסיווג מאפיינים לפי 5 השכנים הקרובים ביותר. דוגמה לפלט הרצה של שלב האימון ניתן לראות ב (איור 16). ראוי לציין את מהירות קידוד הייצוג של התמונות. על כך ארחיב בסעיף (10.2).

```
dir the database...done!
[14-Dec-2012 20:00:36] LLC image representation - start
..... [14-Dec-2012 20:01:06] 100 images processed
..... [14-Dec-2012 20:01:33] 200 images processed
..... [14-Dec-2012 20:02:03] 300 images processed
….
..... [14-Dec-2012 20:49:38] 8900 images processed
..... [14-Dec-2012 20:50:14] 9000 images processed
..... [14-Dec-2012 20:50:48] 9100 images processed
.....[14-Dec-2012 20:51:06] LLC image representation - end
```

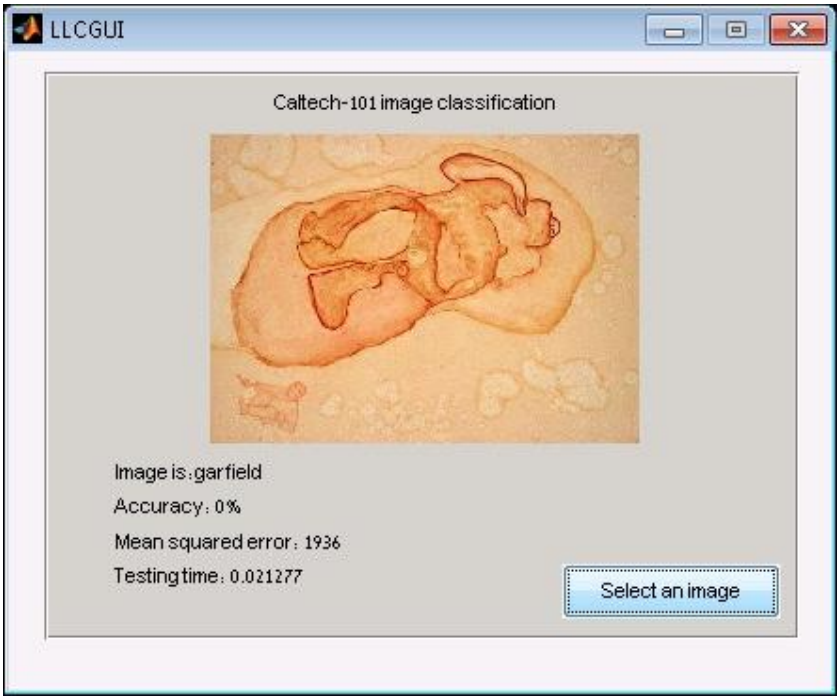
איור 16 – פלט הרצה של שלב הפקת המאפיינים ממאגר התמונות Caltech-101

בשלב הבא, בונים מסווג שמוחזר כמודל חיזוי של הייצוגים של 101 האובייקטים על ידי תיוג כל אובייקט עם הייצוג שלו (המסווג מבוסס למידה מונחית). מודל זה משמש לסיווג אובייקטים חדשים מאותן 101 המחלקות שנלמדו. התוצרים של שלב זה הם, כאמור, מודל של ייצוגי LLC

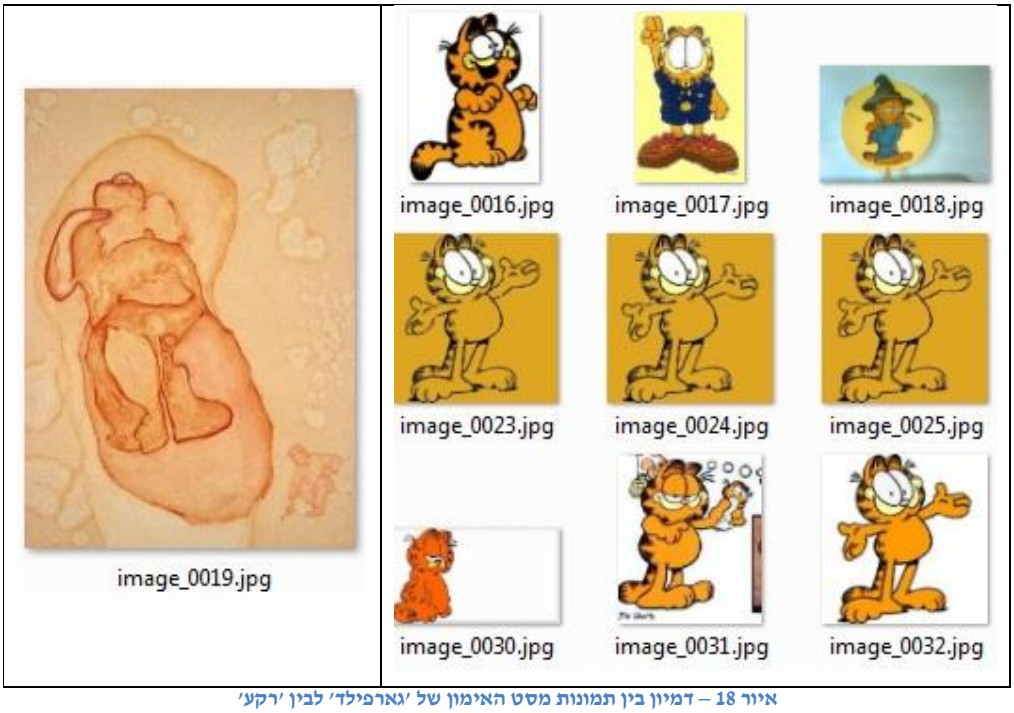
ובנוסף: אוצר מילים (קודים) ואוצר ייצוגי LLC של תמונות. המודל ישמש לחזות בזמן הריצה את ההתאמה המרבית בין דוגמת הקלט לבין אחת מהמחלקות שנלמדו.

השלב העיקרי השני הוא שלב *הסיווג* והוא מתבצע "בזמן אמת" (במצב מקוון). בשלב הראשון, מפיקים מאפיינים מהתמונה שהמשתמש בחר בדיוק באותה הצורה כמו בשלב האימון. בשלב הבא, יוצרים ייצוג LLC של התמונה. בשלב הבא, טוענים את המודל שנוצר בשלב האימון ומעבירים אותו עם ייצוג LLC דליל של התמונה כפרמטרים לפונקציית החיזוי (predict). פונקציה זו מחזירה את התיוג של האובייקט (לפי התיוג משלב הלמידה) ואת מידת הדיוק והשגיאה של המודל. למשל ב (איור 14), פונקציית החיזוי מחזירה את התווית 'Faces', דיוק 100%, ושגיאה ממוצעת 0.

לעיתים יש גם שגיאות של מודל החיזוי ובמקרים אלו מוחזרים ערכי השגיאה ומידת הדיוק. כפי שניתן לראות ב (איור 17), השגיאה בדרך כלל מאוד הגיונית. במקרה זה, ניתן לראות שיש דמיון בין תמונת הרקע מסט האימון (איור 18 משמאל) לבין המודל שמייצג תמונות של גארפילד. במקרה זה, תמונת הרקע כתומה ובה קווי מתאר גליים ומעוגלים בגוונים של חום, צהוב וכתום ולעומתה התמונות השונות של גארפילד מסט האימון שגם הוא דמות עגולה עם גוונים של חום וכתום. מה-"טעויות" הללו ניתן להסיק שיש הגיון רב בפעולת סיווג ונכונות המודל שעומד מאחוריה.



איור 17 – דוגמה לשגיאה בסיווג 'רקע' ל - 'גארפילד'



איור 18 – דמיון בין תמונות מסט האימון של 'גארפילד' לבין 'רקע'

10.2 יתרונות האלגוריתם הלכה למעשה

ערכתי מספר ניסויים כדי להשוות את הביצועים של LLC ו ScSPM על מאגר התמונות Caltech-101. את הניסויים ערכתי על מכונה עם זיכרון פיזי בנפח 8 גיגה ומעבד אינטל Core i8 3.4 גיגה הרץ.

התוצאות מוצגות ב (טבלה 1.10) להלן.

	LLC	ScSPM
זמן בניית ייצוג המאפיינים	50 ד'	510 ד'
גודל ייצוג המאפיינים	34.7 קילו בייט	1,607 קילו בייט
גודל אוצר המילים (dictionary)	565 קילו בייט	17,612 קילו בייט
גודל מודל החיזוי	16 מגה בייט	20 מגה בייט

טבלה 1.10 – תוצאות ביצועים של LLC לעומת ScSPM

כפי שניתן לראות, האלגוריתם LLC יעיל בצורה יוצאת מן הכלל, הן בזמנים והן בנפח הזיכרון שהוא דורש. בהתחשב בעובדה שהדיוק לא נפגע, האלגוריתם בהחלט מוכיח את עליונותו על לא מעט שיטות סיווג נפוצות כמו שניתן לראות ב (פרק 9).

10.3 שיפור המודל

ניתן לשפר את יכולות חיזוי המודל על ידי כמה טכניקת אימון שנקראת אימון-ביקורת-בדיקה (train-validation-test). בשיטה זו, מחלקים את סט התמונות מהם בונים את המודל ל 3 קבוצות בצורה רנדומלית. קבוצה אחת תהיה קבוצת האימון שתכיל בערך 60% מכלל התמונות, קבוצה שניה תהיה קבוצת הביקורת ותכיל 20% מכלל התמונות וקבוצה שלישית תהיה קבוצת הבדיקה ותכיל את 20% התמונות שנותרו. כהערת אגב אומר שהגדלים האלו ניתנים לשינוי וזה תלוי מאוד בסוג הנתונים והבעיה הספציפית, אולם בדרך כלל זה סדר גודל החלוקה.

אחרי שחילקנו את התמונות ל 3 קבוצות, יוצרים מודל מסט האימון ובודקים את השגיאה הממוצעת מול סט הביקורת ולא מול סט הבדיקה. חוזרים על הבדיקה הזו מספר פעמים עד שמגיעים למודל שנותן את השגיאה המינימלית. את המודל הזה בודקים מול הסט השלישי שהוא סט הבדיקה המקורי. בצורה הזו מקבלים שיפור של המודל עד לדרגה מסוימת כך שהבדיקה שלו מתבצעת על מודל מאומן עם סט "חדש" שעליו לא בוצע אימון.

למעשה, התוצאות שמוצגות ב (פרק 9) נבדקו בדרך זו.

11 מסקנות וכיווני מחקר עתידיים

בעבודה זאת סקרתי כמה שיטות חדישות לסיווג אובייקטים באמצעות קידוד דליל ומקומי. שיטות אלו מראות תוצאות יוצאות מן הכלל בביצועי הדיוק, הזיכרון והזמן של הלמידה והסיווג שלהן. כאמור, בזכות ביצועים אלו לשיטות הללו יתרונות מעשיים רבים. החל מסיווג בזמן אמת (מצלמות מעקב לדוגמה) וכלה בסביבת האינטרנט בה גודל מאגרי הנתונים מהווה אתגר משמעותי בסדרי הגודל עבור שיטות רבות.

ישנם כמה כיווני מחקר עתידי לעבודות שסקרתי. אחד, יהיה מעניין לבדוק איך ScSPM תתמודד עם מאפייני SIFT שנדגמו בצורה דחוסה בכמה רזולוציות. שתיים, לנסות את השיטה LLC על ידי מילון שנלמד בצורה (Block Sparse) לפי [44] ולהשוות ביצועים. כותבי העבודה מציעים ללמוד את אוצר המילים בצורה מונחית. עבודה כמו [45] מהווה בסיס למחקר כזה. שלוש, לנסות לחקור/להוכיח בצורה אמפירית למה קל יותר להפריד לינארית מאפיינים המקודדים על ידי קודים דלילים בעזרת פונקציות גרעין לינאריות ו SVM לינארי ובכך לשפר את הייצוג כדי שההפרדה תהיה מקסימלית.

הקידוד הדליל והמקומי הם תחומים חדשים יחסית בעולם הראייה הממוחשבת ועדיין מתבצעת עבודה רבה כדי להביא אותם לתוצאות אופטימליות. לאחרונה, לאור הביצועים של שיטות כמו שסקרתי ולאור היכולות להתמודד בסביבות שמהוות אתגר לשיטות אחרות, מתקיים מחקר רב כדי לנסות ולמצוא שיפורים בדיוק בזיכרון ובמהירות של מודלים המבוססים על קידוד דליל ומקומי. עבודות כמו [44] בה מציעים אלגוריתם הלומד אוצר מילים דליל בצורה מובנית. אוצר מילים דליל מאפשר שימוש יעיל בזיכרון ושיחזור מהיר של המתארים המקוריים. עבודות כמו [35] בהן בעזרת מספר הנחות סבירות על דלילות אוצר המילים והקידוד ניתן לקבל פתרונות יחידים (unique) לבעיית הקידוד. בצורה זו ניתן לבצע שיחזור מדויק יותר של המתארים של התמונה המקורית.
עבודה כמו [34] המראה שיפור בזמני קידוד המאפיינים בעזרת feed-forward network וכך מאפשרת זמני למידה וסיווג חסרי תקדים.
חבילות פיתוח למודלים דלילים כמו [46] מפותחות ופתוחות לקהל הרחב ונראה כי תחום זה תופס תאוצה ונראה בו עוד מחקר רב בשנים הקרובות.

נספח א מכונת וקטורים תומכים

רקע

מכונת וקטורים תומכים (Support Vector Machine - SVM), היא טכניקה של למידה מונחית. אלגוריתם למידה חישובית זה הוצג על ידי ולדימיר ופניק בשנת 1963, ומאז מהווה כלי מרכזי בפתרון בעיות באמצעים סטטיסטיים. כנהוג בתחום זה, דוגמאות האימון מיוצגות כווקטורים במרחב לינארי. תכליתו של שלב האימון היא בניית מסווג (classifier) אשר מפריד בצורה מדויקת ככל האפשר בין דוגמאות אימון חיוביות ושליליות. המסווג שנוצר ב SVM הוא מפריד לינארי אשר יוצר מרווח גדול ככל האפשר בינו לבין הדוגמאות הקרובות לו ביותר מתוך קבוצות האימון.

על מישור מפריד אופטימלי

נניח ש $1\leq \mathrm{i} \leq \mathrm{N}$ (x_i, y_i) הם סט של זוגות אימון כך שלכל דוגמה $x_i \in R^d$, d מימד מרחב הקלט, ישנה תווית $y_i \in \{1, -1\}$. המטרה הסופית הנה להגדיר על מישור אשר מחלק את סט הדוגמאות הללו כך שכל הנקודות עם אותה התווית נמצאות באותו הצד של העל מישור. מכאן שצריך למצוא w ו- b כך ש:

(1)

$$\mathrm{y_i}(\mathrm{w}\cdot \mathrm{x_i} + \mathrm{b}) > 0, \qquad \mathrm{i} = 1, \dots, N$$

אם קיים על מישור המספק את (1), כלומר על מישור המפריד לינארית את סט הדוגמאות החיוביות והשליליות, אז ניתן להגדיר את הסט *כניתן לחלוקה ליניארית*. במקרה זה, מחשבים את w ו- b כך:

(2)

$$\min_{1\leq i\leq N} y_i(w \cdot x_i + b) \geq 1, \qquad i = 1, \dots, N$$

כלומר, שהמרחק בין הנקודה הקרובה ביותר לעל מישור יוגדר כ $\frac{1}{\|w\|}$. מבין כל העל מישורים המפרידים, האחד אשר בעבורו המרחק לנקודה הקרובה ביותר מסט האימון הוא מקסימאלי יקרא *על מישור מפריד אופטימאלי* - Optimal Separating Hyperplane OSH. היות שהמרחק לנקודה הקרובה ביותר הנו $\frac{1}{\|w\|}$, משמע שמציאת ה- OSH שווה למינימיזציה של $\| w \|^2$ תחת האילוצים (2). המרווח $\frac{2}{\|w\|}$ נקרא שוליים, ואלו ה- OSH הנו העל מישור המפריד אשר ממקסם את השוליים. ניתן לראות בשוליים מדד ליכולת הכללה: ככל שהשוליים גדולים יותר, כך נצפה להכללה טובה יותר. היות ש $\| w \|^2$ הנו קמור, ניתן למצוא את המינימום שלו תחת אילוצים ליניאריים בעזרת כופלי לגראנג'.

אם נסמן באמצעות $a = (a_1, \dots, a_N)$ את N כופלי לגראנג' הלא שליליים אשר מקושרים לאילוצים (2), בעיית האופטימיזציה שלנו מסתכמת במקסום:

(3)

$$\mathrm{W}(a) = \sum_{i}^{N} a_i - \frac{1}{2} \sum_{i,j=1}^{N} a_i a_j y_i y_j x_i \cdot x_j$$

תחת האילוצים: $\sum_{i=1}^{N} y_i a_i = 0$, $a_i \geq 0$.

את בעיית האופטימיזציה ניתן לפתור באמצעות תכנות ריבועי.

כאשר וקטור $a^0 = (a_1^0, \dots, a_N^0)$ אשר הנו הפתרון של בעיית המקסימום קיים, ניתן להרחיב את ה- OSH באופן הבא:

(4)

$$w_0 = \sum_{i=1}^{N} a_i^0 y_i x_i$$

וקטורים תומכים הם הנקודות עבורן $a_i^0 > 0$ מקיים את (2) עם שוויון.

בהתחשב בהרחבה (4) של w_0 , פונקציית ההחלטה של העל מישור יכולה להיכתב כ:

(5)

$$f(x) = sgn\left(\sum_{i=1}^{N} a_i^0 y_i x_i \cdot x + b_0\right)$$

- **מידע שלא ניתן להפרדה לינארית**

כאשר המידע לא ניתן לחלוקה ליניארית, ניתן לפתור את הבעיה בעזרת משתני עזר: $\varepsilon_1, \cdots, \varepsilon_N$ כאשר $\varepsilon_i \geq 0$ כך ש:

(6)

$$y_i(w \cdot x_i + b) \geq 1 - \varepsilon_i, \quad i = 1, \cdots, N$$

כדי שתהיה אפשרות לדוגמאות אשר מפרות את (2). מטרת המשתנים ε_i הנה לאפשר לנקודות אשר לא סווגו, אך מתאימות ל $\varepsilon_i > 1$. כך ש $\sum \varepsilon_i$ הנו הגבול העליון של טעויות למידה. ה OSH המוכלל אם כך, הנו הפתרון לבעיה שלהלן:

(7)

$$\frac{1}{2} w \cdot w + C \sum_{i=1}^{N} \varepsilon_i$$

הביטוי הראשון צומצם כדי לשלוט בקיבולת הלמידה כמו במקרה אשר ניתן להפרדה ; המטרה של הביטוי השני הנה לשלוט במספר הנקודות אשר לא סווגו. הפרמטר C נבחר על ידי המשתמש בצורה כזאת שככל ש- C גדול יותר, המשקל של השגיאה גדל.

אימון SVM מצריכה לתקן את C ב- (7), תנאי הענישה בגין סיווג שגוי. כשהמידע הנדגם הוא תמונות, רוב הזמן, ממד הקלט גדול ($1000 \leq$) בהשוואה לגודל של סט אימון, כך שהמידע שמשמש לאימון בדרך כלל ניתן להפרדה ליניארית.

כתוצאה מכך, ערכו של C במקרה זה הנו בעל השפעה מועטה בלבד על הביצועים.

- **SVM לא ליניארי**

המידע אשר מגיע מהקלט ממופה לתוך *מרחב מאפיינים* רב ממדי דרך מספר מיפויים לא ליניאריים. המישור OSH נוצר בתוך מרחב מאפיינים זה. אם נחליף את X במיפוי שלו בתוך מרחב המאפיינים $\varphi(x)$, (3) נהיה:

$$\mathrm{W}(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \cdot \alpha_j \cdot y_i \cdot y_j \cdot \varphi(x_i) \cdot \varphi(x_j)$$

אם יש לנו $\mathrm{K}(x_i \cdot x_j) = \varphi(x_i) \cdot \varphi(x_j)$, אז רק K נדרש בתוך אלגוריתם האימון ואז לא משתמשים במיפוי φ מפורשות לעולם. בניגוד לזה, בהינתן גרעין סימטרי וחיובי, לפי התיאוריה של מרסר [47] מצביעה על כך שכן קיים מיפוי φ כך ש $\mathrm{K}(x, y) = \varphi(x) \cdot \varphi(y)$.

ברגע שהגרעין K אשר עונה על תנאי מרסר נבחר, אלגוריתם האימון מורכב ממציאת מינימום

(8)

$$\mathrm{W}(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \cdot \alpha_j \cdot y_i \cdot y_j \cdot K(x_i, x_j)$$

ופונקציית ההחלטה הופכת ל:

(9)

$$f(x) = sgn(\sum_{i=1}^{N} a_i \cdot y_i \cdot K(x_i, x) + b)$$

- **סיווג מספר מחלקות**

אלגוריתם SVM עוצב לצורך סיווג בינארי. כשמתמודדים עם כמה מחלקות, כגון זיהוי עצמים וסיווג תמונות, יש צורך בשיטה שתדע לטפל במספר מחלקות ביחד. מבין האפשרויות קיימות האופציות הבאות:

- להתאים את אלגוריתם SVM כדי לשלב בו את הלמידה הרב מחלקתית ישירות בתוך אלגוריתם לפתרון בעיות בינאריות.

- לשלב כמה מסווגים בינאריים: "אחד מול אחד" משמע לבצע השוואת זוגות של סיווגים אחד מול השני בעוד "אחד מול האחרים" משמע לבצע השוואה בין סיווג ספציפי מול כל השאר.

הביצועים של שתי השיטות הם כמעט זהים. לכן נבחר את השיטה הפחות מורכבת: "אחד מול האחרים".

באלגוריתם "אחד מול האחרים", n על מישורים נוצרים, כאשר n הנו מספר הסיווגים. כל על מישור מפריד בין סיווג מסוים אחד לבין האחרים. בשיטה זו, מקבלים n פונקציות החלטה $1 \leq k \leq n$ (f_k) מסוג (5). סיווג של נקודה חדשה x ניתן באמצעות:

$$argmax_k \; f_k(X)$$

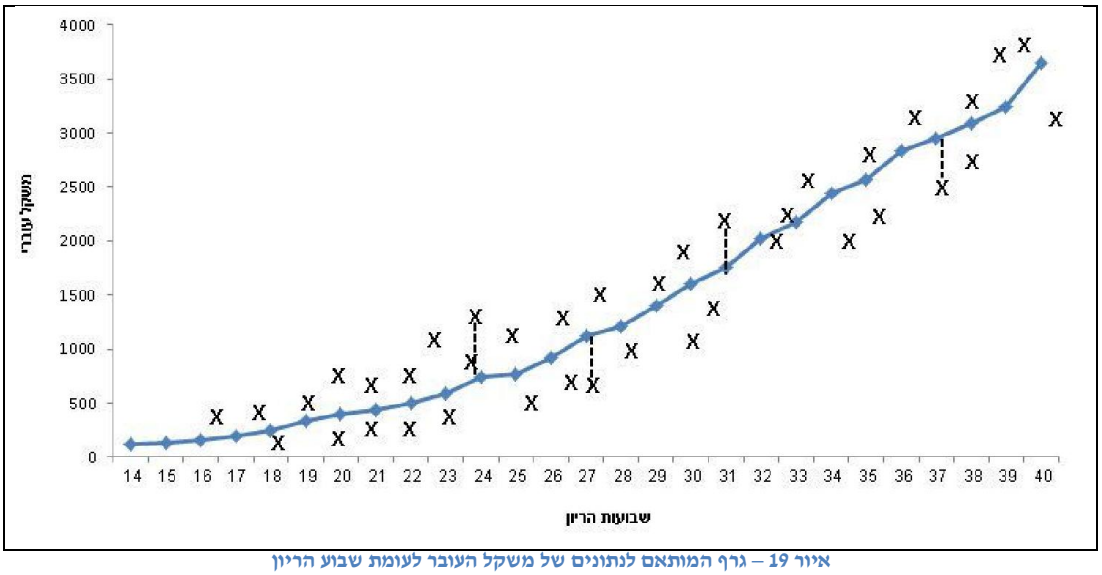
כלומר, סיווג עם פונקציית ההחלטה הגדולה ביותר.

נספח ב שיטת הריבועים הפחותים

שיטה זו נקראת גם "מינימום ריבועים" (Least Squares, Least Mean Squares, Least Squares Fitting). השיטה שימושית כאשר מנסים למצוא מודל מנבא או קשר בין נתונים שמתוארים בדרך כלל כווקטורים מממד מסוים על גרף. לרוב משתמשים בפונקציות המתאימות תוצאה לערכים השונים של הנתונים הנלמדים או ליחס ביניהם.

המטרה בשיטה זו היא למצוא פונקציה לכל הנקודות המייצגות ווקטורים במרחב רב ממדי מסוים, כך שהמרחק בין ערכי הפונקציה לבין הנקודות הנתונות יהיה מינימלי ככל הניתן. הפונקציה יכולה לעבור דרך כל הנקודות ויכולה שלא. אגב, אם נתאים פונקציה שתעבור דרך כל הנקודות אנו עלולים לקבל מודל מנבא לא טוב שכן הוא מותאם יותר מדי לסט האימון ופחות לדוגמאות בדיקה חדשות מחוץ לסט האימון. בעיה זו נקראת "התאמת יתר" (over-fitting problem).

דוגמה למשל יכולה להיות סט X של נתונים המייצגים ערכים של משקל עובר וסט Y של נתונים המייצגים ערכים של שבוע מתחילת הריון. אוספים את המידע ממדגם כלשהו ומציגים אותו על גרף הבנוי משני צירים לפי הסטים X ו Y. בשלב הבא מנסים למצוא פונקציה שמתקרבת ככל הניתן לכל נקודה על הגרף (איור 19).



איור 19 – גרף המותאם לנתונים של משקל העובר לעומת שבוע הריון

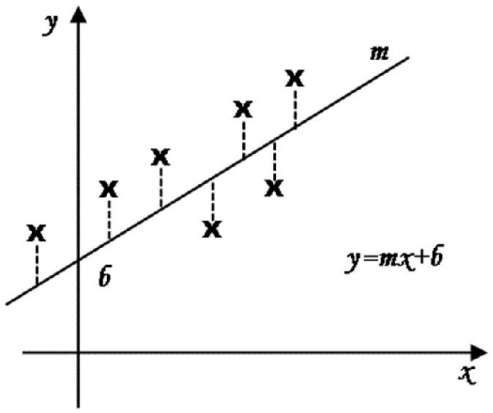
תיאור מתמטי של הבעיה

יהיו $x_1 x_2, \dots, x_n$ נתונים כלשהם.

לפי שיטת הריבועים הפחותים צריך להתאים פונקציה $y = f(\vec{x}, \vec{a})$ לסידרה של נתוני אימון $(y_i, \overrightarrow{x_i})$ כאשר $\vec{a}$ הוא אוסף של פרמטרים בצורה הבאה:

$$\min_{\vec{a}} \sum_{i=1}^{n} (y_i - f(\vec{x_i}, \vec{a}))^2$$

כלומר המטרה למצוא את הפרמטרים $\vec{a}$ שיביאו את הסכום למינימום. ב (איור 20) ניתן לראות התאמה של פונקציה לינארית לפיזור הנתונים X.



איור 20 – דוגמה לרגרסיה לינארית

הישר y = mx + b מותאם לנקודות X על ידי צמצום המרחק האנכי בין כל נקודה x_i לישר.

נספח ג K השכנים הקרובים ביותר

K השכנים הקרובים ביותר (K-nearest-neighbor (KNN הוא אלגוריתם סיווג המסווג דוגמת קלט לפי רוב של שכנים הקרובים אליה.

בהינתן קבוצת לימוד בעלת n אברים מתויגים לקבוצות שונות ונתונה דוגמת קלט חדשה לסיווג. האלגוריתם ימצא התאמה בין דוגמת הקלט ל k האיברים ה"מתאימים" אליו ביותר מתוך n האברים המתויגים. התאמה יכולה להיות המרחק האוקלידי בין דוגמת הקלט לאברים השונים. דוגמת הקלט תסווג לקבוצת האברים השכיחה ביותר מבין k האברים שנמצאו.

נספח ד K ממוצעים

אלגוריתם מציאת K אשכולות ממוצעים (k-means clustering) משמש ללמידה לא מונחית.

תיאור אלגוריתם

- המטרה: לצמצם את סכום ריבועי המרחקים האוקלידיים בין וקטורים x_m ומרכזי האשכולות הקרובים אליהם ביותר v_k.

$$\min_{V} \sum_{m=1}^{M} \min_{k=1\ldots K} \|x_m - v_k\|^2$$

- האלגוריתם:
  - נאתחל k מרכזי אשכולות
  - עד שנגיע להתכנסות בערכי מרכזי האשכולות נבצע את האיטרציה הבאה:
    - נשייך כל וקטור x_m ל v_k הקרוב ביותר אליו.
    - נחשב את k מרכזי האשכולות החדשים כממוצע המרחקים החדש בין כל הווקטורים המרכיבים כל אשכול.

ביבליוגרפיה

[1] A. Shashua, Y. Gdalyahu, and G. Hayun, "Pedestrian detection for driving assistance systems: Single-frame classification and system level performance," in *Intelligent Vehicles Symposium, 2004 IEEE*, 2004, pp. 1–6.

[2] J. Markoff, "Google cars drive themselves, in traffic," *The New York Times*, vol. 10, p. A1, 2010.

[3] J. Ponce, "Toward category-level object recognition," 2006.

[4] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Workshop on statistical learning in computer vision, ECCV*, 2004, vol. 1, p. 22.

[5] D. Liu, D. Sun, and Z. Qiu, "Bag-of-Words Vector Quantization Based Face Identification," 2009, pp. 29–33.

[6] M. Nishimura, S. Scherf, and M. Behrmann, "Development of object recognition in humans," *F1000 Biology Reports*, vol. 1, Jul. 2009.

[7] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, 2006, vol. 2, pp. 2169–2178.

[8] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," *Computer Vision and Image Understanding*, vol. 106, no. 1, pp. 59–70, 2007.

[9] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International journal of computer vision*, vol. 88, no. 2, pp. 303–338, 2010.

[10] G. Griffin, A. Holub, and P. Perona, "The Caltech-256," 2007.

[11] K. Yu, Y. Gong and T. H. J. Yang, "Linear Spatial Pyramid Matching Using Sparse Coding for Image Classification," Miami, FL, USA,, 2009.

[12] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 2010, pp. 3360–3367.

[13] J. Sivic and A. Zisserman, "Efficient Visual Search of Videos Cast as Text Retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 4, pp. 591–606, Apr. 2009.

[14] J. Puzicha, J. M. Buhmann, Y. Rubner, and C. Tomasi, "Empirical evaluation of dissimilarity measures for color and texture," in *Computer Vision, 1999. The*

Proceedings of the Seventh IEEE International Conference on, 1999, vol. 2, pp. 1165–1172.

[15] M. Mohri, *Foundations of machine learning*. Cambridge, MA: MIT Press, 2012.

[16] K. Grauman and T. Darrell, "The pyramid match kernel: Efficient learning with sets of features," *The Journal of Machine Learning Research*, vol. 8, pp. 725–760, 2007.

[17] J., Marszalek, M., Lazebnik, S. and C. Z. Schmid, "Local features and kernels for classification of texture and object categories: a comprehensive study.," vol. 73(2):213–238., 2007.

[18] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, 1991, pp. 586–591.

[19] G. Csurka, C. R. Dance, F. Perronnin, and J. Willamowski, "Generic visual categorization using weak geometry," in *Toward Category-Level Object Recognition*, Springer, 2006, pp. 207–224.

[20] E. Nowak, F. Jurie, and B. Triggs, "Sampling strategies for bag-of-features image classification," *Computer Vision–ECCV 2006*, pp. 490–503, 2006.

[21] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[22] K and C. M. Schmid, "Scale and affine invariant interest point detectors.," vol. 60:63–86., 2004.

[23] F and A. S. Zisserman, "Viewpoint invariant texture match-ing and wide baseline stereo.," Vancouver, Canada.,, 2001.

[24] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, 2005, vol. 1, pp. 886–893.

[25] V. Chandrasekhar, G. Takacs, D. M. Chen, S. S. Tsai, Y. Reznik, R. Grzeszczuk, and B. Girod, "Compressed histogram of gradients: A low-bitrate descriptor," *International Journal of Computer Vision*, vol. 96, no. 3, pp. 384–399, 2012.

[26] T. Kadir and M. Brady, "Saliency, scale and image description," *International Journal of Computer Vision*, vol. 45, no. 2, pp. 83–105, 2001.

[27] K. Mikolajczyk and C. Schmid, "An affine invariant interest point detector," *Computer Vision—ECCV 2002*, pp. 128–142, 2002.

[28] C. Harris and M. Stephens, “A combined corner and edge detector,” in *Alvey vision conference*, 1988, vol. 15, p. 50.

[29] C. C. Lee and K. Y. Chu, “CUDA-accelerated Hierarchical K-means.”

[30] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, “Object retrieval with large vocabularies and fast spatial matching,” in *Computer Vision and Pattern Recognition, 2007. CVPR’07. IEEE Conference on*, 2007, pp. 1–8.

[31] J. Eichhorn and O. Chapelle, “Object categorization with SVM: kernels for local features,” 2004.

[32] O. Chapelle, P. Haffner, and V. N. Vapnik, “Support vector machines for histogram-based image classification,” *Neural Networks, IEEE Transactions on*, vol. 10, no. 5, pp. 1055–1064, 1999.

[33] H. Lee, A. Battle, R. Raina, and A. Y. Ng, “Efficient sparse coding algorithms,” *Advances in neural information processing systems*, vol. 19, p. 801, 2007.

[34] K. Kavukcuoglu, M. A. Ranzato, and Y. LeCun, “Fast inference in sparse coding algorithms with applications to object recognition,” *arXiv preprint arXiv:1010.3467*, 2010.

[35] C. J. Hillar and F. T. Sommer, “Ramsey theory reveals the conditions when sparse coding on subsampled data is unique,” *arXiv preprint arXiv:1106.3616*, 2011.

[36] K. Yu, T. Zhang, and Y. Gong, “Nonlinear learning using local coordinate coding,” *Advances in Neural Information Processing Systems*, vol. 22, pp. 2223–2231, 2009.

[37] H. Zhang, A. C. Berg, M. Maire, and J. Malik, “SVM-KNN: Discriminative nearest neighbor classification for visual category recognition,” in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, 2006, vol. 2, pp. 2126–2136.

[38] O. Boiman, E. Shechtman, and M. Irani, “In defense of nearest-neighbor based image classification,” in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, 2008, pp. 1–8.

[39] P. Jain, B. Kulis, and K. Grauman, “Fast image search for learned metrics,” in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, 2008, pp. 1–8.

[40] J. van Gemert, J. M. Geusebroek, C. Veenman, and A. Smeulders, “Kernel codebooks for scene categorization,” *Computer Vision–ECCV 2008*, pp. 696–709, 2008.

[41] L. Fei-Fei and P. Perona, “A bayesian hierarchical model for learning natural scene categories,” in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, 2005, vol. 2, pp. 524–531.

[42] P. Over, G. M. Awad, J. Fiscus, M. Michel, A. F. Smeaton, and W. Kraaij, “TRECVID 2009-goals, tasks, data, evaluation mechanisms and metrics,” 2010.

[43] R. E. Fan, K. W. Chang, C. J. Hsieh, X. R. Wang, and C. J. Lin, “LIBLINEAR: A library for large linear classification,” *The Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.

[44] K. Rosenblum, L. Zelnik-Manor, and Y. C. Eldar, “Dictionary optimization for block-sparse representations,” in *AAAI Fall 2010 Symposium on Manifold Learning*, 2010, pp. 50–58.

[45] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, “Supervised dictionary learning,” *arXiv preprint arXiv:0809.3083*, 2008.

[46] Willow Project, *SPArse Modeling Software*. Inria.

[47] J. Mercer, “Functions of positive and negative type, and their connection with the theory of integral equations,” *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, pp. 415–446, 1909.

Abstract

Visual object categorization is one of the most challenging issues in computer vision. The difficulty is to identify multi forms and looks of different objects which belong to the same category yet to avoid wrong diagnostics.
For instance, it is not accurate to classify a cat as a dog. Even though both cat and dog share many mutual characteristics such as a tail, a mustache, four legs, pointy ears, fur etc. Each characteristic is slightly different in spite of the similarities. Moreover, there are also inner class variations – Poodle ears are different from Labrador ears.
Therefore, a good classification method should be able to identify a variety of different objects which belong to the same category, yet not to misidentify lookalike objects from different categories.
There are few more problems which make the classification even harder such as the image quality (different lighting, distortion), occlusion (a person which is partly hidden by a tree), and background clutter (chameleon on a tree) etc. Such problems make it even harder to classify and require different methods and solutions.
Above all that, to be able to classify effectively, visual information should be represented compactly and uniquely as possible and still preserve the sampled information.
Nowadays there are many projects and applications which use visual object categorization technologies. Autonomic cars which navigate without human interference through miles of difficult scenery, mini UAV, satellites, car alerting systems in case of road hazards or accidents and augmented reality applications are only a partial list of projects and applications which use visual object categorization technologies.
Most of the projects nominated above, are not working flawlessly. There is a vast research in the field of object categorization going on to find new and improved methods. There are still many problems which need to be solved so that those projects will work better. Some of the problems and suggested solutions will be discussed in this paper.

Table of Contents

The Open University of Israel
Department of Mathematics and Computer Science

Sparse and Local Coding Methods for Classification of Visual Information

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
Eyal Dahari

Prepared under the supervision of Dr. Tal Hassner

February 2014