זיהוי אוטומטי של ציפורים על פי הקלטות ציוצים

פרויקט בבינה מלאכותית 236502 אביב 2019

דו״ח מסכם

דניאל קרנצדורף 205503725 ועופרי רביבו 311398937

מבוא

מטרת הפרויקט הנה ליצור מסווג מלאכותי של זני ציפורים לפי הקלטות ציוציהן, ולהדגים כיצד הוא פועל.
ניתן לחלק את הפקת הצליל של ציפורים לשני קטגוריות עיקריות:

- שירה (Song) - הפקת צליל ארוכה ומורכבת בגיוון המוזיקלי שלה. לרוב מתבצעת על ידי זכרים לטובת חיזור או הגנה טריטוריאלית.
- קריאה (Call) - הפקת צליל קצרה לטובת תקשורת עבור אזהרה, מנוסה, תיאום בתעופה וכו׳.

באופן כללי, ניתן לחלק את זיהוי הציוצים למספר מרכיבים עיקריים, עליהם כל מסווג, אנושי או מלאכותי, יצטרך לתת את הדעת:

- גבה הקול (Pitch) - ציוצי הציפורים מכסים את כל טווח התדרים הניתנים לשמיעה, אך לכל זן ציפור מספר תדרים אופייניים[1]. תתכן חפיפה בתדר האופייני בין הזנים, כשלעתים התדרים האופיינים יהיו בטווח רחב, לצד וריאציות בתוך כל זן[2], מה שמקשה להתבסס על גבה הקול בלבד לטובת הזיהוי.
- מפעם (Tempo) - קצב הציוץ, כלומר מספר התווים או ההברות, ואורך ההפסקות ביניהם, ביחידת זמן. הטמפו בעל שונות יחסית גבוהה, לעתים בעקבות מטרות ציוץ שונות, אך זיהוי תבניות ריתמיות הוא מפתח משמעותי לזיהוי.
- איכות ההקלטה - שילוב של ציוצי ציפורים שונות ורעשי רקע יחדיו עלול להקשות על הזיהוי.

[1] Moghal et al. - Bird calls frequency distribution analysis to correlate with complexity of syrinx (https://www.mutagens.co.in/jgb/vol.04/6/040608.pdf)
[2] Pitch, and bird song identification (https://www.sibleyguides.com/bird-info/the-basics-of-identifying-bird-sounds/pitch-and-bird-song-identification)

מאגר הנתונים

האתר [3](Xeno-canto (Sharing bird sounds from around the world הינו מאגר שיתופי המכיל מאות אלפי קבצי שמע שהוקלטו על ידי חובבי צפרות ברחבי העולם. כל קובץ מכיל תיוג זן הציפור, ארץ ההקלטה ואופי הציוץ. ההקלטות נתרמו לאתר ברובן המכריע על ידי חובבים פרטיים באיכות גבוהה, ולא על ידי גופי מחקר.

בחלק משמעותי מן ההקלטות, הציפור המתויגת נשמעת רק בקטע קצר מתוך ההקלטה, בעוד שחלקים אחרים כוללים קטעי שקט או רעש שאינם רלוונטיים. זאת, בדומה להקלטה אותנטית שמסווג ייאלץ להתמודד עימה במבחן המציאות. בחלק מן ההקלטות אף נשמעים ציוצי מספר ציפורים שונות בו זמנית, כאשר התיעוד על כך מופיע ברשומות המאגר.

כדי למקד את המשימה, הגדרנו לעצמנו כמטרה לסווג בהצלחה בין עשרה זני הציפורים בעלי מספר ההקלטות הרב ביותר. כך, הצטמצמנו למאגר של כ- 25 אלף הקלטות באורך כולל של כ-670 שעות, ובמשקל של 70 GB. ניתן לראות שבין הזנים נמצאים שני צמדים של ציפורים מתתי זנים שונים השייכים לאותו הזן, על מנת "לאתגר" את המסווג.

0 ירגזי כחול אירואסייתי

1 אדום החזה

2 פרוש מצוי

3 צלוב-מקור מצוי

42 עלווית החורף

5 עלווית אפורה

6 סבכי שחור-כיפה

7 שחרור

2 קיכלי רונן

9 ירגזי מצוי

[3] https://www.xeno-canto.org

על מנת לעבד כל הקלטה ביעילות חישובית ככל הניתן, ומתוך הבנתנו את אורך הקטעים הסביר לקלט עבור מסווגים נפוצים, החלטנו לחתוך כל הקלטה לאורך של ארבע שניות בלבד. זאת, על ידי יוריסטיקה פשוטה ככל הניתן:
תחילה הורדנו את קצב הדגימה ל- 20,000 דגימות לשניה (מאפשר תדרים עד 10 קילו הרץ), לאחר מכן העברנו את השמע דרך High pass filter שמסנן החוצה את כל התדרים מתחת ל-700 הרץ, על מנת לא לקחת בחשבון רעשי רקע בתדרים שלא אופייניים לציפורים אותם החלטנו לסווג. לבסוף חתכנו כל קטע מקורי ל-4 שניות סביב הנקודה בעלת העוצמה המקסימלית בהקלטה המסוננת, כאשר הסבירות הגבוהה ביותר היא שבנקודה זו נשמעת הציפור בבירור, ולא רעש רקע אחר. כך, הגענו לייצג כל הקלטה כוקטור בעל 80,000 כניסות.

כאשר משתמשים במסווג לטובת Inference, ניתן לחתוך הקלטות ארוכות יותר מגודל ה-Frame הנ״ל לקטעים בשיטת Overlapping sliding window, וקבלת סיווג על כל קטע בנפרד, ולבצע אגרגציה על כלל תוצאות הסיווג.

גישות לזיהוי וניתוח שמע

כיום, נראה כי מרבית הניסיון לסיווג קטעי שמע הושג בתחומי זיהוי קולי וסיווג קטעי סאונד סביבתיים מהמרחב האורבני. במסגרת זו, נחשפנו לשתי גישות עיקריות[5][4] לסיווג אודיו -

- למידה ישירות מייצוג גולמי של אות השמע (end-to-end) כוקטור חד מימדי בציר הזמן, על ידי שכבות קונבולוציה חד מימדיות. יתרון גישה זו טמון במספר פרמטרים לאימון יחסית נמוך, ובכך שלמידה ישירות ממימד הזמן אמורה להתאים לחילוץ Features מתפישת המקצב של הציוצים. עם זאת, את ניתוח התדרים הרשת אמורה ללמוד בעצמה.
- ייצוגי ביניים כדוגמת ספקטרוגרמות, שהינן ייצוג דו-מימדי מעובד של ספקטרום התדרים ביחס לזמן. שיטה זו פופולרית יותר, ומתאימה כקלט לרשתות מאומנות מוכרות לעיבוד תמונה. יתרונה בכך שמידע אודות טווח התדרים מובא לרשת כקלט, כך שהיא אינה צריכה ללמוד אותו בעצמה.

[4] Lonce Wyse - Audio spectrogram representations forprocessing with Convolutional Neural Networks (https://arxiv.org/pdf/1706.09559.pdf)
[5] Youness Mansar - Audio Classification : A Convolutional Neural Network Approach (https://github.com/CVxTz/audio_classification)

למידה ישירות מגל הקול במימד הזמן

תחילה, מימשנו בעצמנו ב- Pytorch את הארכיטקטורה המוצעת[6] על ידי החוקרים Abdoli, Cardinal & Koerich מאוניברסיטת קוויבק, אשר נבנתה במקור למטרת Environmental sound classification, והשיגה 82 אחוזי דיוק על מאגר UrbanSound8k.
בחרנו בגרסה המתאימה לקלט באורך ארבע שניות. בעוד שהרשת המקורית אומנה על קבצי השמע אשר נדגמו מחדש לתדירות של 16,000 דגימות בשנייה, אנו השתמשנו בה על קלט הדגום ב- 20,000 דגימות לשניה, על מנת לא לאבד תדרים נחוצים עבור סוג הדאטא שלנו.

גרסה זו מורכבת משזירה של חמש שכבות קונבולוציה חד-מימדיות עם אקטיבציית ReLU, ושלוש שכבות Sub-sampling מסוג Max-pooling. אחרי כל אקטיבציה מופעלת שכבת Batch Normaliztion, המנרמלת את פלט שכבת האקטיבציה (feature map) על מנת למנוע overfitting הנובע מעוצמת ההקלטה המקורית בדוגמאות האימון. הגרנולריות של שכבת הקונבולוציה הראשונה זהה בכל הגרסאות המוצעות, כך שגודל כל פילטר הינו 64.

לבסוף, את הסיווג מבצעות ארבע שכבות Fully-Connected, וביניהן מבוצע Drop-out אשר מאפס כל אלמנט בפלט השכבה הקודמת בהסתברות 0.25, גם כן על מנת להתמודד למול overfitting. השכבה האחרונה מכילה 10 נוירונים, כמספר התגיות בקבוצה שלנו, וה- Feature map אותו היא יוצרת מוזן לפונקציית SoftMax אשר מנרמלת אותו על מנת ליצור התפלגות הסתברות בין התיוגים. בהתאם לסוג הפלט הנ״ל, השתמשנו בפונקציית ההפסד Cross entropy loss.

את אימון הרשת ביצענו באמצעות אלגוריתם חיפוש Gradient descent מסוג Adadelta[7], אשר מבצע כוונון דינאמי של ה- Learning rate בזמן האופטימיזציה.

---

[6] Abdoli et al. - End-to-End Environmental Sound Classification using a 1D Convolutional Neural Network (https://arxiv.org/pdf/1904.08990v1.pdf)
[7] Matthew D. Zeiler - ADADELTA: An Adaptive Learning Rate Method (https://arxiv.org/abs/1212.5701)

```
TweetNet(
  (layer1): Sequential(
    (0): Conv1d(1, 16, kernel_size=(64,), stride=(2,))
    (1): ReLU()
    (2): BatchNorm1d(16, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (3): MaxPool1d(kernel_size=8, stride=8, padding=0, dilation=1, ceil_mode=False)
  )
  (layer2): Sequential(
    (0): Conv1d(16, 32, kernel_size=(32,), stride=(2,))
    (1): ReLU()
    (2): BatchNorm1d(32, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (3): MaxPool1d(kernel_size=8, stride=8, padding=0, dilation=1, ceil_mode=False)
  )
  (layer3): Sequential(
    (0): Conv1d(32, 64, kernel_size=(16,), stride=(2,))
    (1): ReLU()
    (2): BatchNorm1d(64, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  )
  (layer4): Sequential(
    (0): Conv1d(64, 128, kernel_size=(8,), stride=(2,))
    (1): ReLU()
    (2): BatchNorm1d(128, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  )
  (layer5): Sequential(
    (0): Conv1d(128, 256, kernel_size=(4,), stride=(2,))
    (1): ReLU()
    (2): BatchNorm1d(256, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (3): MaxPool1d(kernel_size=4, stride=4, padding=0, dilation=1, ceil_mode=False)
  )
  (classifier): Sequential(
    (0): Linear(in_features=2048, out_features=1024, bias=True)
    (1): ReLU()
    (2): Linear(in_features=1024, out_features=256, bias=True)
    (3): ReLU()
    (4): Dropout(p=0.25)
    (5): Linear(in_features=256, out_features=64, bias=True)
    (6): ReLU()
    (7): Dropout(p=0.25)
    (8): Linear(in_features=64, out_features=10, bias=True)
    (9): Softmax()
  )
)
```

מבנה הרשת

ניסויים

את הרשת הראשונה אימנו פעמיים, כאשר בפעם הראשונה הזנו לרשת מכל ההקלטה את האות המקורי הנקי לאחר החיתוך לארבע שניות, ובפעם השנייה את האות המפולטר, לאחר שהורדנו ממנו את התדרים מתחת ל-700 הרץ. בשתי הפעמים הרשת אומנה על 90% מהדוגמאות למשך 100 epochs (כלומר הרשת ראתה כל דוגמה מאה פעם), ב- Batch size של מאה דוגמאות. לאחר מכן בחנו את ביצועי הרשת על 10% הדוגמאות הנותרות, כאשר התייחסנו רק לתיוג הסופי, זה שזכה להסתברות הגבוהה ביותר בהתפלגות.

	אות נקי	אות מפולטר
0 ירגזי כחול אירואסייאתי	81.93%	79.73%
1 אדום החזה	73.14%	82.40%
2 פרוש מצוי	84.70%	81.03%
3 צלוב-מקור מצוי	76.81%	67.63%
4 עלווית החורף	86.31%	88.27%
5 עלווית אפורה	83.66%	84.15%
6 סבכי שחור-כיפה	76.99%	82.62%
7 שחרור	83.09%	76.81%
8 קיכלי רונן	67.91%	63.10%
9 ירגזי מצוי	70.76%	76.79%
כולל	**78.52%**	**78.80%**

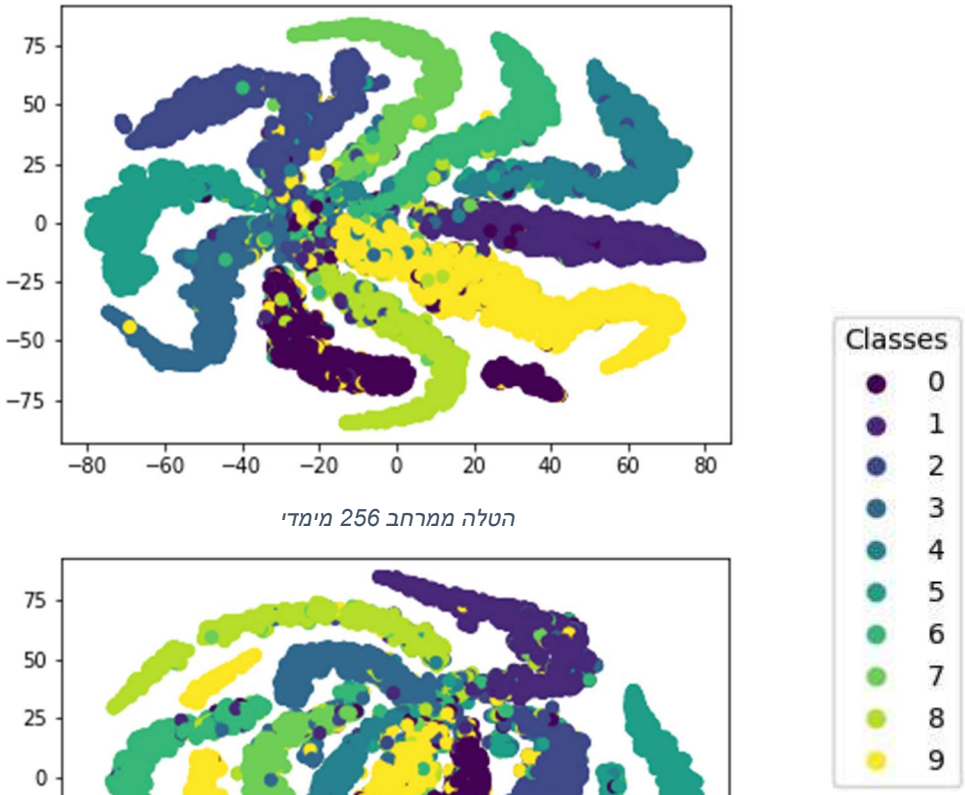
רשת ראשונה - אחוזי הצלחה לפי זן

ניתן לראות שהרשת שפעלה על האות המפולטר זכתה לאחוזי הצלחה מעט גבוהים יותר, בעיקר לאור עלייה ניכרת בדיוק סיווג של זן "אדום החזה" (1).

בהמשך הדו"ח ננסה לנתח את תוצאות אלו ולהבין את יכולת ההפרדה של המסווג בין הזנים השונים, וכיצד שכבות הקונבולוציה השונות במסווג המאומן פועלות.

ניתוח הרשת

כאמור, לרשת ה-FC שאחראית לסיווג יש שני hidden layers בגודל 256 ו-64 נוירונים. תחילה, ניסינו לבצע החזיה (Visualization) של הגיאומטריה של מרחב התכונות (Feature space) שאותו הרשת יוצרת בעזרת שתי שכבות אלו. העברנו כל ציוץ קדימה דרך המסווג, והטלנו כל נקודה ב- Feature maps ה-256 וה-64 מימדי למרחב הדו-מימדי, באמצעות אלגוריתם הורדת מימד לא-לינארי בשם t-SNE.



הטלה ממרחב 256 מימדי

הטלה ממרחב 64 מימדי

ניתן לראות לפי ה- Clustering בגרף ששלב חילוץ ה-Features מאפשר יכולת ההפרדה בין הזנים השונים, ובפרט, הפרדה בין תתי זנים קרובים (4 ו-5, 0 ו-9), אשר אינם קרובים באופן חריג אחד לשני במרחב הדו-מימדי.

Predicted class / Actual class	0	1	2	3	4	5	6	7	8	9
0	X	0.24	0.12	0.02	0.07	0.02	0.02	0.07	0.00	0.41
1	0.12	X	0.10	0.07	0.03	0.03	0.07	0.07	0.17	0.33
2	0.02	0.16	X	0.06	0.16	0.04	0.08	0.08	0.14	0.26
3	0.00	0.06	0.08	X	0.06	0.02	0.15	0.15	0.15	0.33
4	0.02	0.14	0.14	0.10	X	0.07	0.07	0.05	0.12	0.29
5	0.06	0.12	0.21	0.03	0.18	X	0.09	0.00	0.06	0.24
1	0.02	0.16	0.08	0.06	0.10	0.06	X	0.14	0.12	0.24
7	0.14	0.14	0.11	0.03	0.11	0.00	0.06	X	0.17	0.23
8	0.10	0.15	0.05	0.08	0.03	0.00	0.15	0.30	X	0.13
9	0.29	0.13	0.21	0.09	0.08	0.02	0.06	0.05	0.07	X

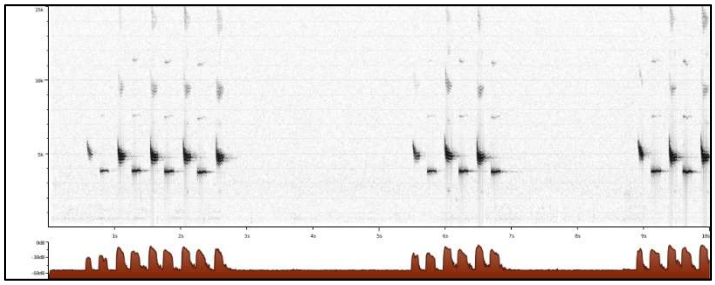
אחוז סיווגים שגויים לכל זן (Confusion Matrix)

מבדיקת תוצאות המסווג על הסיווגים השגויים, מצאנו שעבור רוב הזנים הסיווג השגוי הנפוץ ביותר היה סיווג כזן מספר 9 (ובהתאם, זהו הזן עבורו אחוזי הדיוק של המסווג הנמוכים ביותר). ואכן, בגרף הרלוונטי לשכבה המאוחרת יותר ממרחב התכונות ה- 64- מימדי, ניתן לראות שזן מספר 9 תופס חלק גדול במרכז הגרף, וחותך את כל הזנים פרט לזן מספר 8.

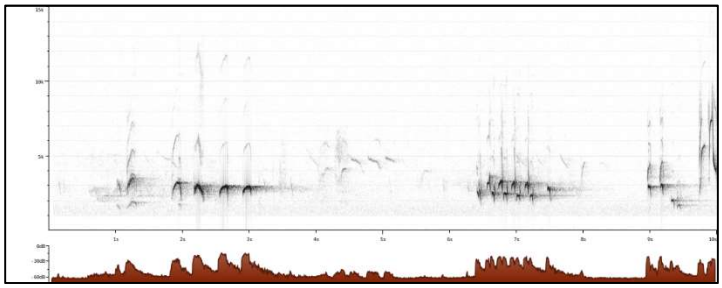
בניסיון להבין תוצאה זו, ביצענו בדיקה ידנית על מנת לאפיין את התדרים הנפוצים של ציוצי הזנים השונים:

Species	0	1	2	3	4	5	6	7	8	9
Frequency	6-8 KHz	8KHz	4KHz	3-5 KHz	3-5 KHz	3-7 KHz	3-4 KHz	6-8 KHz	2-3 KHz	5KHz

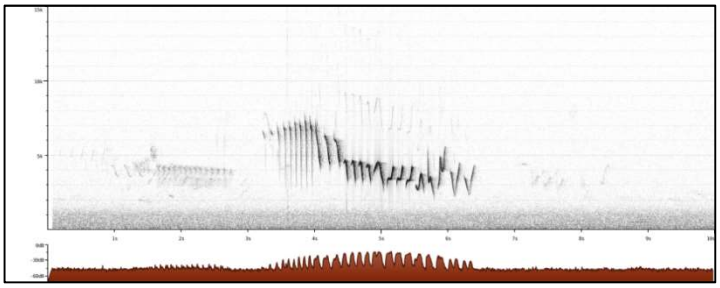
ניתן לראות כי זן מספר 9 מפיק תדר של 5KHz, אשר מייצג בקירוב את ממוצע התדרים של הזנים השונים, דבר אשר תומך בתוצאות שהתקבלו. ניתן לראות כי זן מספר 8 מפיק את התדרים הנמוכים ביותר, ועובדה זו יכולה להסביר את ה"התרחקות" שלו במישור התכונות מזן מספר 9.



Parus Major (9)



Turdus philomelos (8)

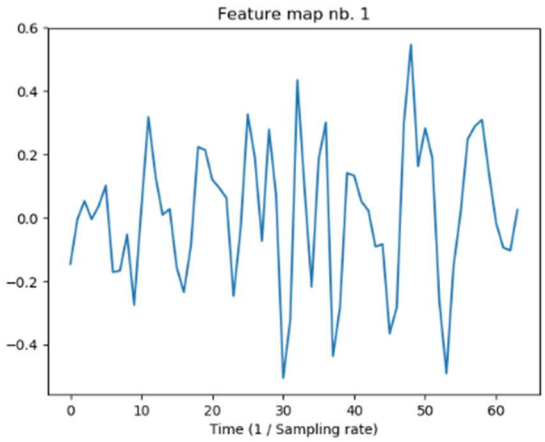


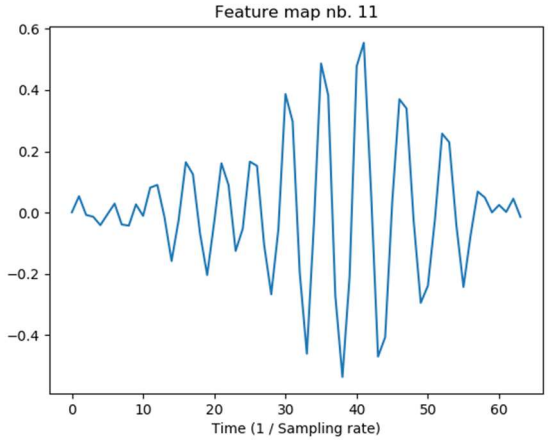
Phylloscopus trochilus (5)

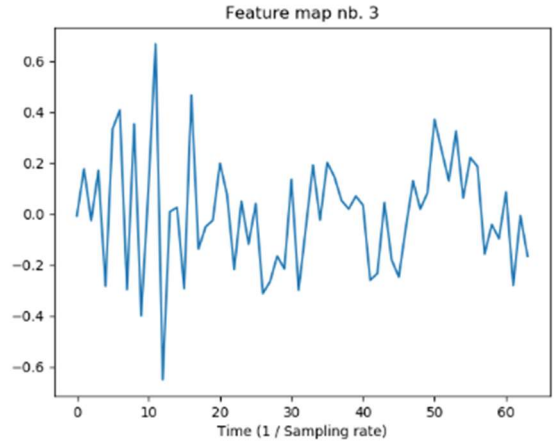
ניתוח שכבת הקונבולוציה הראשונה

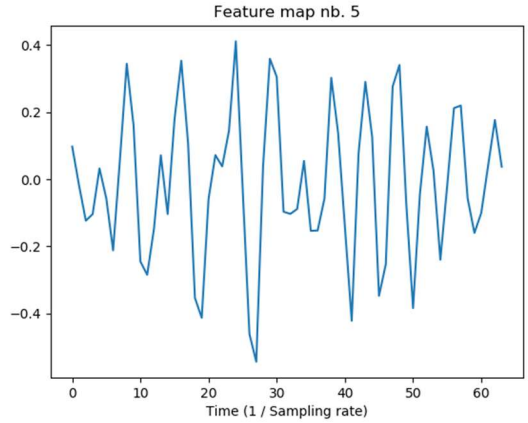
שכבת הקונבולוציה הראשונה מורכבת מ-16 פילטרים בגודל 64 כ״א. ייחודה של שכבה זו שהיא פועלת ישירות על השמע - כל פילטר ״נע״ לאורך וקטור השמע ומבוצעת מכפלה פנימית בין השמע לבין המשקולות של הפילטר. מכפלה פנימית זו מקבלת ערך מקסימלי כאשר וקטור השמע זהה לוקטור המשקולות, כלומר ניתן לראות בהתבוננות על הפילטרים ״מה הרשת מחפשת״ ישירות מהשמע.

להלן מספר דוגמאות של הפילטרים על ציר הזמן:







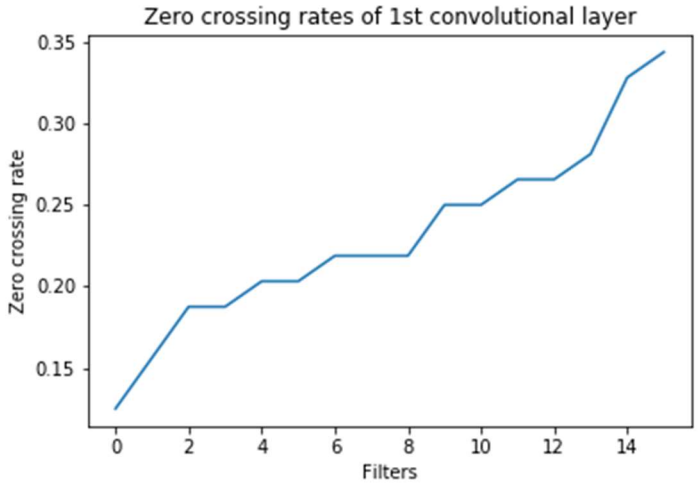


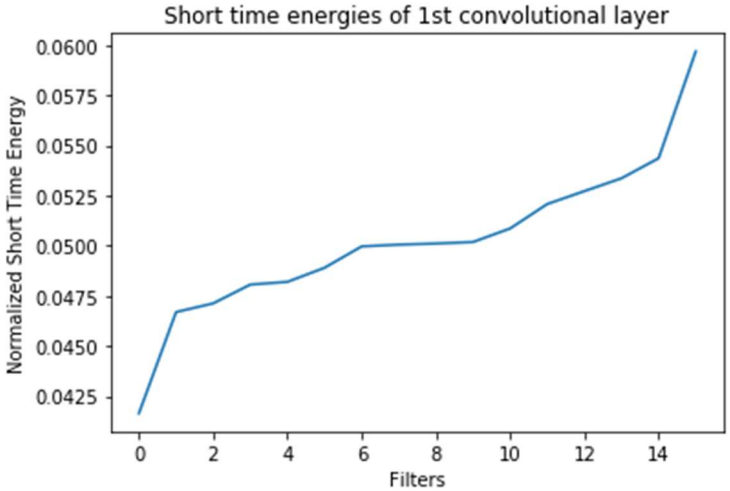
באופן גס, ניתן להבחין בשונות בין הפילטרים על ציר הזמן. בפילטר מספר 11 האמפליטודה דועכת במהירות משני צדידי המקסימום, באופן המתאים לקריאת ציפור קצרה, לעומת פילטר מספר חמש בו גובה האמפליטודה יחסית קבוע, באופן שניתן לייחס לשיר.

שני Time domain features נפוצים בתחום ה- Speech detection ו- music information retrieval הינם:

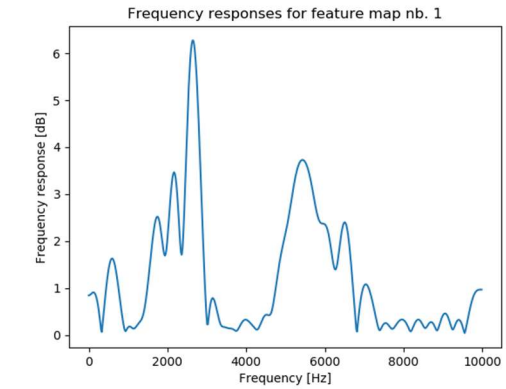
- Zero crossing rates - מודד בפשטות את מספר הפעמים שהאמפליטודת הקול חוצה את הערך אפס. מניתוח הפילטרים עולה ZCR יחסית גבוה, המתאים לאות רועש או High pitch.
- Short time energy - מודד את השונות בגובה המקסימלי של אמפליטודת האות. מהניתוח עולה אנרגיה ממוצעת נמוכה יותר מזו של קול אנושי.

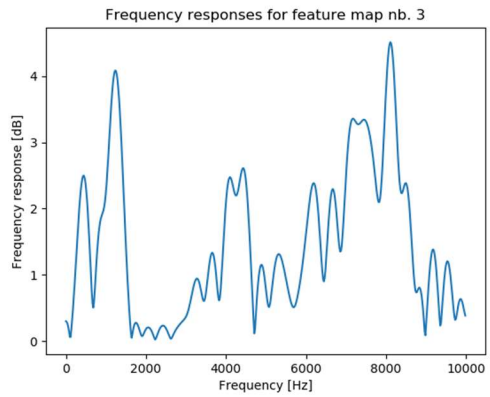
יש לציין שבדרך כלל מודדים את האנטרופיה של מאפיינים אלו, כלומר השינוי בהם לאורך ציר הזמן. עם זאת, אין אנו יכולים להציע ניתוח כזה בעקבות האורך הקצר של הפילטרים בשכבה זו.

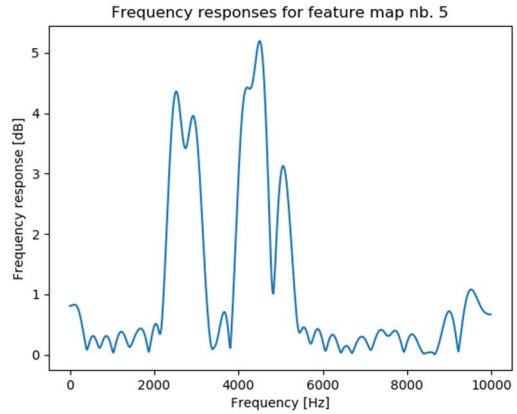


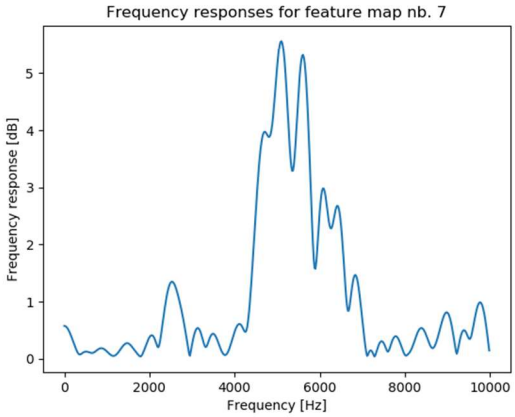


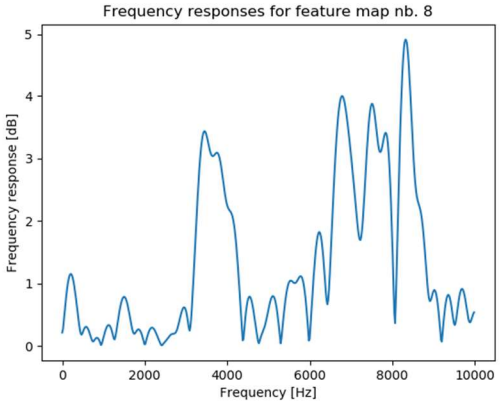
לאחר מכן, ביצענו FFT לכל פילטר כדי למצוא את פירוק לתדרים:

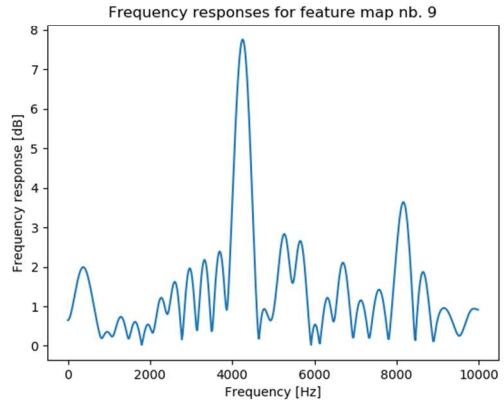


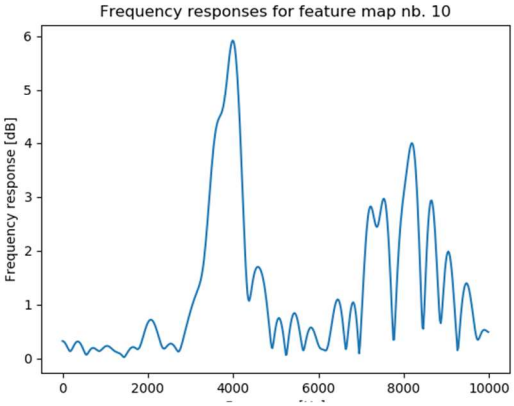


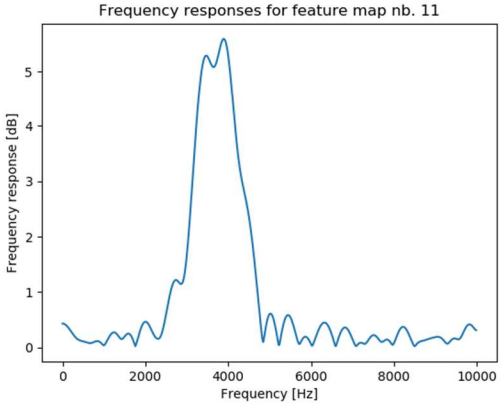


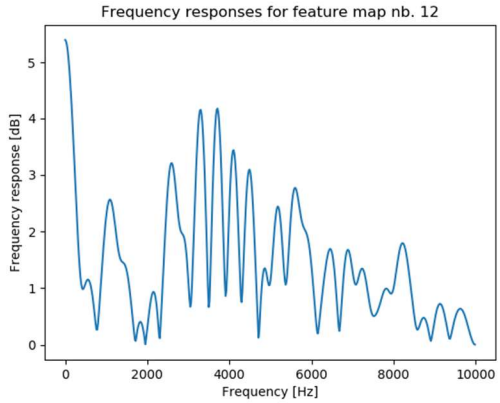


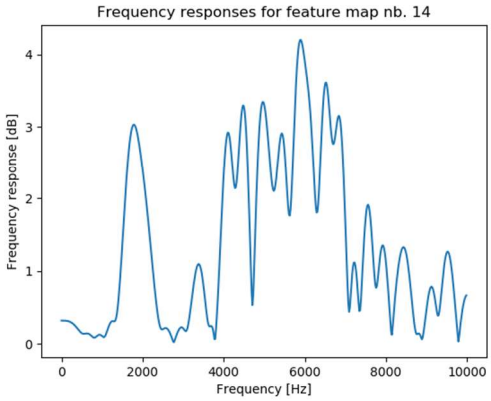


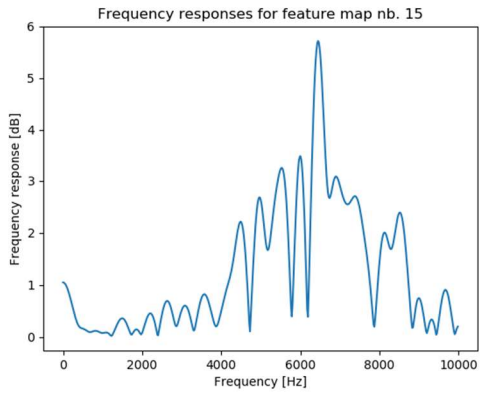


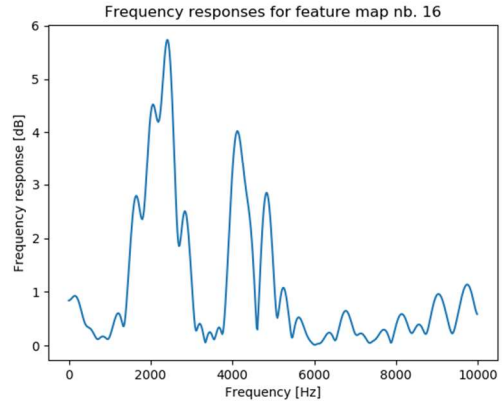












ניתן להסיק ששכבת הקונבולוציה הראשונה מתפקדת בין היתר כ- Bandpass filter. כלומר מעבירה תדרים בטווח מסוים ודוחה את אלו שמחוצה לו. חלק מהפילטרים פעילים בעיקר בטווח תדרים מצומצם, רובם באזור ה-2 ו-4 קילו הרץ, אך סך טווח התדרים המכוסה על ידי כלל הפילטרים אכן מכסה את הטווח האופייני לציוצי הציפורים שלנו, כולל פילטרים הרגישים לסף הגבוה של הטווח (8 קילו הרץ):

פילטר	תדר אופייני (קילו הרץ)
1	2,6
2	3
3	1,8
4	קשה להכריע
5	3,4
6	3,6
7	5
8	4,7,9

פילטר	תדר אופייני (קילו הרץ)
9	4
10	4,8
11	4
12	<1
13	3
14	2,6
15	7
16	2,4

ניתוח שאר שכבות הקונבולוציה

בניגוד לשכבת הקונבולוציה הראשונה, שכבות הקונבולוציה הבאות אינן מקבלות כקלט את וקטור השמע המקורי אלא את מפת האקטיבציה של השכבה הקודמת, ועל כן לא ניתן להסתפק בניתוח ישיר של המשקלים.

כדי להדגים כל פילטר בנפרד, השתמשנו באלגוריתם Random restart gradient ascent אשר מחפש וקטור שמע במרחב הקלטים האפשריים לרשת, עבורו האקטיבציה הממוצעת של אותו הפילטר היא הגבוהה ביותר. טכניקה זו נועדה להמחיש כיצד פילטר מסוים מגיב ל – raw audio waveform אותו אנו מזינים כקלט לרשת המאומנת. זאת, בהשראת הטכניקה "Deep dream"[8] שמעדכנת קלט מסוים באמצעות גרדיאנט של ציון של class ספציפי, כדי להחזות את אותה המחלקה על הקלט.

את הפונקציה אותה אנו מחפשים למקסם קבענו במקרה זה כממוצע האקטיבציה של הפילטר ביחס לקלט ההתחלתי. בכל איטרציה אנו מבצעים "guided"[9] backpropagation באמצעות הספרייה keras-vis של גרדיאנט הפונקציה, כלומר מפעפעים אחורה רק את הגרדיאנטים האי שליליים, כיוון שאנו מתעניינים לחזק בקלט רק את ה- Features שהפילטר מזהה, ולנטרל את השפעת הדחייה של מאפיינים שאינם

[8] AUDIO DEEPDREAM: Optimizing raw audio with convolutional networks – Ardilla et al. from Google Brain
[9] Springenberg et al - Striving for simplicity: The all convolutional net (https://arxiv.org/pdf/1412.6806.pdf)

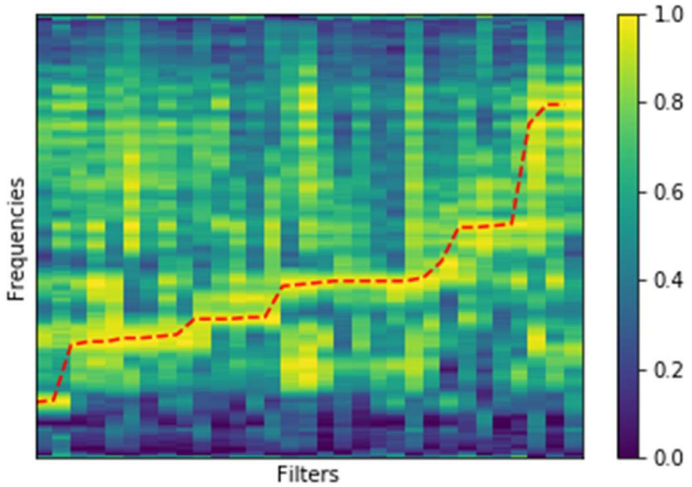
מזוהים בוקטור באותה איטרציה. לבסוף, אנו מנרמלים את הגרדיאנט, על מנת שגודל הצעד בכל איטרציה יהיה זהה, ומוסיפים את הגרדיאנט המנורמל לקלט.

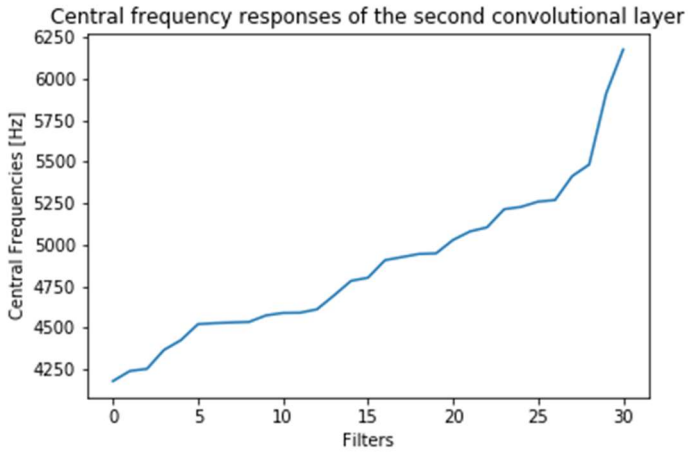
את החיפוש אנו מתחילים מרעש לבן אקראי בהתפלגות אחידה בקטע (0.015 ,0.015-], כיוון שהוא מכיל את כל ספקטרום התדרים, ובכך מגדיל את הסבירות שהאקטיבציה אותה אנו מעוניינים להדגים תופעל.
בפועל ראינו שעבור פילטרים מסויימים החיפוש נוטה להתקע כבר בתחילתו, ולכן ניסינו לבצע את החיפוש ממספר רב של נקודות התחלה.

בכתיבת האלגוריתם התבססנו על מימוש דומה של ספריית Keras בתחום הראייה הממוחשבת[10]. בהחזיית שכבות קונבולוציה מתחום זה ניתן להבחין בעלייה במורכבות התכונות הנלמדות עם התקדמות בעומק הרשת (קווים, צורות פשוטות ודמויות). לעומת זאת, כתוצאה ממגבלות חוש השמיעה האנושי, בתחום השמע קיים קושי לתפוש ולהדגים את הטקסטורות הנלמדות על ידי הרשת, ובפרט הפרדה שלהן להיררכיה לפי מורכבותן. לכן, הסתפקנו ב"ניתוח קר" של תוצאות החיפוש, בו חיפשנו באופן מוטה מופעים של תכונות תדריות מוכרות.

כאשר ביצענו ניתוח תדרים לוקטורי השמע שהתקבלו, בו חיפשנו את התדר המרכזי של כל פילטר, ראינו שעבור שכבת הקונבולוציה השנייה התקבלה כצפוי פריסת תדרים רחבה בטווח 4-6 kHz, תוך דחיית התדרים בקצוות הסקאלה:

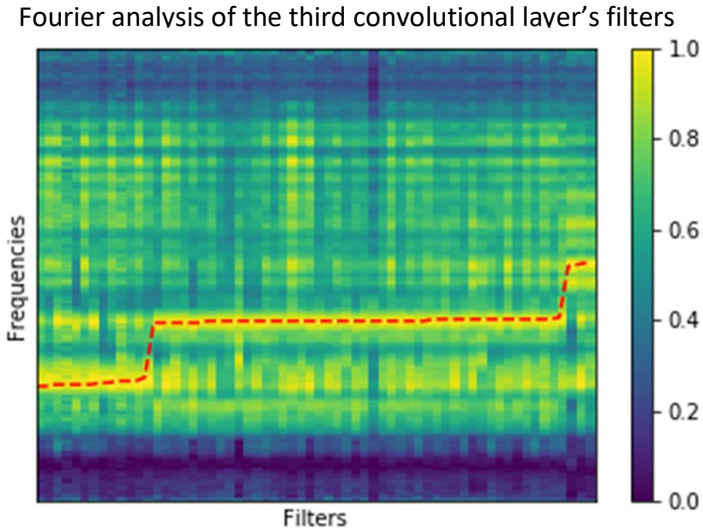
Fourier analysis of the second convolutional layer's filters

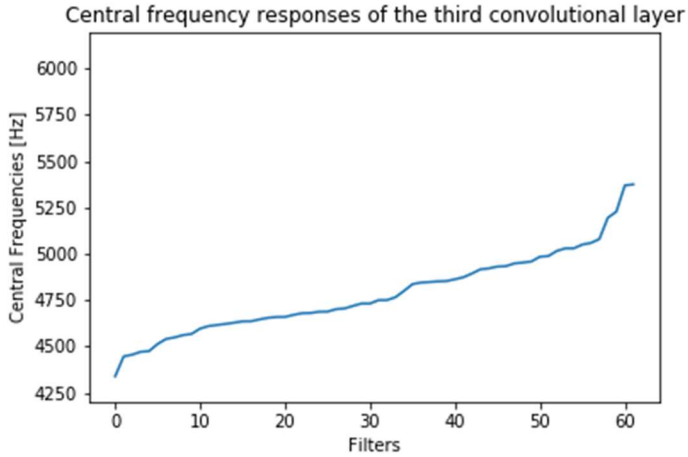




[10] Visualization of the filters of VGG16, via gradient ascent in input space (https://keras.io/examples/conv_filter_visualization)

לעומת זאת, עבור שכבת הקונבולוציה השלישית ואילך, קיבלנו תגובה תדרית קבועה בפס מצומצם ברוחב 1kHZ :

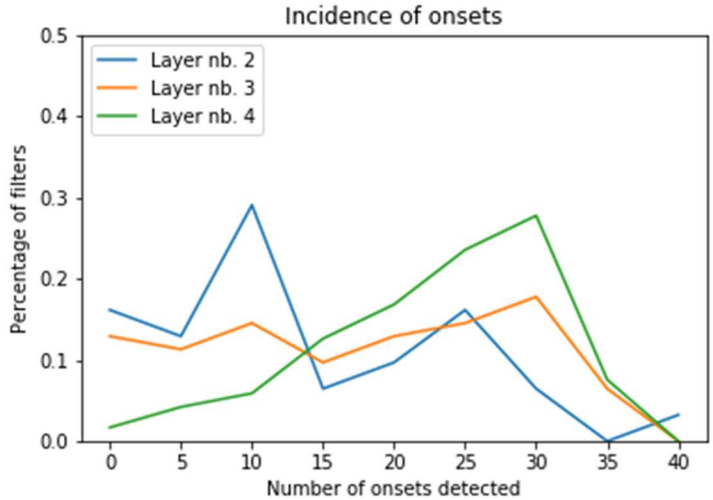




ההסבר נובע מכך שהקלט עליו הפילטרים פועלים בשכבות קונבולוציה אלו הוא עיבוד לא-לינארי של הכניסות, בו כניסות בתדרים שאינם אופייניים נדחים כבר על ידי שכבת הקונבולוציה הקודמות. ככל שמעמיקים בשכבות, העיוות הלא-לינארי גדל.

לכן, בעוד שבשתי השכבות הראשונות ניתן היה לזהות, בקירוב, תפקוד כ- filterbank, שהינו מטבעו לינארי, לא ניתן לצפות שבפילטרים הלא לינארים בשכבות העמוקות יותר תהיה משמעות תדרית קלאסית ביחס לכניסה המקורית.

בנוסף, בדקנו בקלטים אליהם הגענו בחיפוש את כמות מופעי ה- Onsets, שהינה תכונה של עלייה חדה באנרגיה, המופיעה בתחילת הפקת הצליל בלבד. מהניתוח עולה שקיימים יותר מופעים של התכונה בקלטים שמתאימים לאקטיבציות בשכבות הקונבולוציה המאוחרות, שמזהים תכונות מורכבות יותר, ולכן ייתכן שלתכונה זו יכולת הפרדה מסוימת בין הזנים.



למידה מספקטרוגרמות

ספקטרוגרמה היא ייצוג של השמע המקורי כ-Feature map דו-מימדי, כאשר לציר הזמן נוסף ציר התדר, בעוד שהערכים מייצגים את האמפליטודה.
אנחנו השתמשנו בספקטרוגרמה מסוג Mel, שהנה פופולרית בתחום למידת המכונה, והשיגה את התוצאות הטובות ביותר[11], ביחס לסוגי ספקטרוגרמות אחרים, על מאגר UrbanSound8k, עליו נבחנה במקור הרשת הקודמת.

לאחר ביצוע Short-time Fourier transform, טווח התדרים ממופה לסולם Mel.
הטרנספורמציה הלא-לינארית נקבעה כך שהמרחק בין כל שני קטעי שמע בסולם Mel נשמע לאוזניים אנושיות במרחק זהה. זאת, בניגוד לסולם הרץ, בו קל להבחין בהבדל בתדרים נמוכים אך יחסית קשה להבחין בין תדרים גבוהים באותו המרחק.

עבור כל קטע הקלטה במשך ארבע שניות יצרנו ספקטרוגרמה באמצעות הספרייה Librosa[12]. בעזרתה ביצענו STFT לכל תת קטע בוקטור באורך 2048, בדילוגים של 256 כניסות (כך שתתי הקטעים חופפים), וחילקנו את ציר התדרים ל-128 קטעים בסולם Mel. באמצעות תמונות הספקטרוגרמה אימנו את רשת הקונבולוציה VGG[13] במימוש של הספרייה torchvisions.

הרשת המאומנת השיגה 76.86 אחוזי הצלחה על קבוצת המבחן, כלומר מעט פחות מדויקת מהרשת הראשונה הלומדת ישירות מוקטור השמע. על פניו ניתן להסביר זאת על ידי איבוד מידע כתוצאה מהקטנת ציר הזמן לטובת יצירת ציר התדר, בעוד שציר התדר ככל הנראה אינו נושא תוספת אינפורמציה איכותית למסווג. למול הסבר אפשרי זה אימנו בשנית את הרשת, אך הפעם ביצענו את הטרנספורמציה בדילוג של 128 כניסות, כלומר הכפלנו את ציר הזמן מ-313 ל626 כניסות, וכך השווינו את מספר הכניסות המוזנות לרשת לגודל הקלט של הרשת הראשונה. עם זאת, ביצועי הרשת לא השתפרו.

כמו כן, לטובת יצירת הספקטרוגרמה הדו-מימדית נלקח בחשבון רק ה- magnitude של התוצאה המרוכבת של התמרת הפורייה, בעוד שמידע על הפאזה של הגל הושמט, וייתכן שמידע זה חיוני לבעיית הסיווג שלנו, בדומה לבעיות בתחום ה- Speech processing[14].

הסבר אפשרי נוסף הינו שהעומק הדרוש לרשתות עיבוד תמונה, ובפרט הרשת אותה אימנו, גורם למרחב הפרמטרים ללמידה לתפוח למול ה-scale המצומצם של ה-dataset שלנו, וכן עלול לגרום לבעיית vanishing gradient בה הלמידה בשכבות הראשונות זניחה בסוף תהליך ה-backpropagation.

---

[11] Muhammad Huzaifah - Comparison of Time-Frequency Representations for Environmental Sound Classification using Convolutional Neural Networks (https://arxiv.org/pdf/1706.07156.pdf)
[12] https://librosa.github.io/librosa/generated/librosa.feature.melspectrogram.html
[13] Karen Simonyan, Andrew Zisserman - Very Deep Convolutional Networks for Large-Scale Image Recognition (https://arxiv.org/abs/1409.1556)
[14] Mowlaee et al. - INTERSPEECH 2014 Special Session: Phase Importance in Speech Processing Applications (https://www.isca-speech.org/archive/archive_papers/interspeech_2014/i14_1623.pdf)

כדי לשלול את השפעת בחירת המסווג הספציפי על התוצאות, אימנו על אותו הקלט את הרשת Resnet-152[15], שהנה בעלת מבנה דומה של גודל קרנל 3x3 בכל שכבות הקונבולוציה, אך מכילה כחצי מכמות הפרמטרים הנלמדים (60M לעומת 143M), באמצעות טכניקה של residual modules. למרות זאת, ביצועי הרשת נפלו במקרה זה ל-70.8% דיוק.

סיכום וכיוונים להמשך

במהלך הפרויקט הצלחנו לבנות רשת לסיווג ציוצי הציפורים אשר הצליחה לעמוד בציפיות הראשוניות שלנו לאחוזי דיוק. עם זאת, בניגוד לצפיותינו, ניסיונות לשיפור הדיוק כגון שימוש בספקטוגרמות לא צלחו ואחוזי הדיוק נשארו בטווח ה- 80%.

ביצענו מספר ניסויים ובדיקות על מנת לחקור את פעולת הרשת, את אופי הסיווג של הפילטרים בשכבות השונות וניתוח ה-audio features. אנו רואים את מרבית ניסיונות הויזואליזציה כהצלחה, ורובם עזרו לנו להבין יותר טוב את מבנה הרשת ואת אופי הלמידה שלה.

עם זאת, בעקבות משאבים מוגבלים ברשותנו (זיכרון, מאיץ חישובי אחד, גודל dataset) הגבלנו את משימת המיון לעשרה זנים. היינו רוצים להגדיל משמעותית את מספר הזנים, ולגוון את ה-Data בקטעים שאינם מתויגים כציפורים (רעשי סביבה שונים), כדי להפוך את משימת הסיווג לפרקטית ורלוונטית עבור משתמשים בסביבה מגוונת אמיתית.

בנוסף, היינו מעבים את יכולת העיבוד הראשוני של קבצי השמע הגולמיים על ידי שימוש בכלים נוספים עבור סגמנטציה של קטעי השמע לציוצים, ולהזין מכל הקלטה מספר רב של דגימות עבור אימון המערכת.

בהנתן פתרונות למגבלות אלה, היה ניתן לפתח אפליקצית Mobile לשימוש יום-יומי לחובבי צפרות. בראייתנו, עקב המגבלות הללו, אפליקציה אשר עושה שימוש במסווג אשר פיתחנו לבעיה המוקטנת לא היתה מספקת מענה מספק לצורך זה.

במהלך הפרויקט מצאנו שעיקר העיסוק בסיווג האודיו מתמקד בתחום ה- Speech recognition, ולכן נאלצנו לעשות שימוש בכלים מתחום זה, ומתחום עיבוד תמונה. עובדה זו לרוב היוותה מכשול, כיוון שההקבלה מתחומים אלו לא היתה טבעית לבעיית הסיווג איתה התמודדנו. לכן, היינו מעוניינים לבצע מחקר מעמיק יותר בעצמנו בתחום, על מנת לשפר את יכולת הבנתנו את פעולת המסווג (Feature extraction).

[15] Kaiming He, at al. - Deep Residual Learning for Image Recognition (https://arxiv.org/abs/1512.03385)