ALARM SOUND DETECTION USING TOPOLOGICAL SIGNAL PROCESSING

Tomer Fireaizen *Saar Ron* *Omer Bobrowski*

Viterbi Faculty of Electrical and Computer Engineering
Technion – Israel Institute of Technology, Haifa, Israel

ABSTRACT

We present a novel approach to alarm sound detection using topological data analysis. Our main focus is on proposing a new set of robust features, based on algebraic topology, that are aimed at capturing global structural information about the dynamical system underlying each input signal. In short, we convert each signal into a point cloud and compute its corresponding persistent homology, from which we can extract a variety of useful numerical features. We demonstrate the power of this framework using the UrbanSound8K dataset and show that, by combining topological features with a classical classification method, we achieve state-of-the-art results.

***Index Terms*—** alarm detection, signal classification, topological signal processing, topological data analysis, persistent homology

1. INTRODUCTION

An *alarm sound* is "a loud noise or a signal that warns people of danger or of a problem" (Oxford English Dictionary). The ability to automatically detect alarms in noisy environments is essential in a wide range of settings, for example – providing alerts for the hearing impaired, or detecting ambulance sirens by autonomous cars. This is a challenging task mainly since the temporal and spectral properties of alarm signals have a tremendous variability (see Figure **??**).

Classical approaches to alarm detection use specific features in the time and frequency domains to detect alarms with a well-known waveform, e.g. ambulance siren [**?**, **?**, **?**, **?**]. More modern approaches [**?**, **?**, **?**] train deep convolutional neural network (CNNs) to classify various types of urban sounds, where alarm sound is one of the classes.

Topological data analysis (TDA) promotes the use of qualitative structural information in data and network analysis [**?**, **?**]. The main tool developed and used in this field is called *Persistent Homology*. Briefly, this tool provides multi-scale information about various kinds of "holes" that may appear in the data (see Section **??**). Persistent homology has been successfully applied in a wide variety of data analytic problems (e.g., in robotics [**?**], network analysis [**?**], neuroscience [**?**], and hyperspectral imaging [**?**]). Most relevant to us, is its use in signal processing. Here, one first converts a discrete-time signal into a point cloud using the *sliding window transformation* (see Section **??**), and then computes persistent homology. The intuition is that the topological structure of the point cloud may reveal useful information about the underlying dynamical system. For example, in [**?**, **?**] it was shown that persistent homology is able to detect periodic and quasi-periodic phenomena in signals. This approach already have found several applications [**?**, **?**, **?**, **?**].

OB was supported in part by the Israel Science Foundation, Grant 1965/19. This research was carried out in the Signal and Image Processing Lab (SIPL) at the Technion.

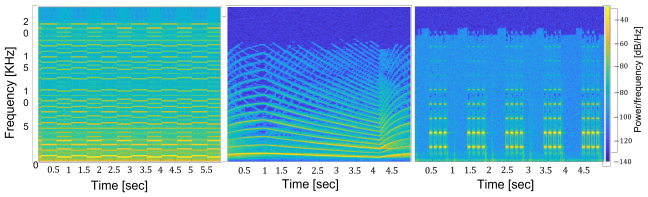

Fig. 1: STFT of three alarm sounds (police siren, ambulance, and alarm clock), demonstrating the variability of the waveforms.

Our goal in this paper is to introduce persistent homology to the task of alarm sound detection. We will show that topological information extracted from a signal significantly enhances detection accuracy, and in fact allows us to achieve state-of-the-art results using a classical classifier. From the TDA point of view there are two main novelties here: (a) We use persistent homology for classifying signals *not* according to any predefined properties (e.g. periodicity); (b) We show that homology in degrees higher than 1, which is rarely used, has in fact a significant contribution to the task at hand. Finally, we note that signal-classification framework we propose here is quite generic and can be applied in many other scenarios.

2. PROPOSED METHOD

The main contribution of this paper is the introduction of topological methods to the task of alarm sound detection. In this section we present the key steps in transforming a time series into a list of topological features. We present it here in general terms, and provide the application-specific details later. We assume that the input (x_n) is a finite discrete-time signal. The key steps are the following:

1. Convert the time series (x_n) into a D-dimensional point cloud $\mathcal{P}_D = \{p_1, \ldots, p_m\} \subset \mathbb{R}^D$. This is done using the *sliding window transformation*. See section **??**.
2. Apply a dimension reduction algorithm to transform $\mathcal{P}_D$ into a lower-dimensional point cloud $\mathcal{P}_d \subset \mathbb{R}^d$, with $d < D$. See section **??**.
3. Use the points in $\mathcal{P}_d$ to construct a *simplicial filtration* $\mathcal{F}$, which can be thought of as an increasing collection of hypergraphs. See section **??**.
4. Compute the persistent homology $\mathrm{PH}_k(\mathcal{F})$ for the filtration generated in step 3, and its corresponding persistence diagram $\mathrm{dgm}_k(\mathcal{F})$. Here k represents the dimension of the topological structures we are interested in. See Section **??**.
5. Extract a list of numerical marginals from the computed persistence diagrams, denote by $T_1, \ldots, T_N$, see Section **??**.

The output of the pipeline above is a list $T_1, \ldots, T_N$ of numerical topological features, representing various measurement related to the

spatial structure of the original signal (x_n) when converted into a point cloud. Next, we will discuss each of the steps above in detail.

2.1. Sliding window transformation

The purpose of the sliding window (SW) transformation [**?**] (also known as 'time-delay embedding') is to convert a discrete-time signal into a point cloud (i.e., a finite collection of points) in a higher dimensional space. The motivation for doing so is the idea that the shape of the point cloud may reveal substantial information about the dynamical system underlying the observed signal. Given a time series (x_n), and two natural numbers D and τ, define

$$p_i = (x_i, x_{i+\tau}, \ldots, x_{i+(D-1)\tau}) \in \mathbb{R}^D.$$

In other words, p_i consists of D samples from the original signal, with interval τ between samples. Next, we define the point cloud $\mathcal{P}_D = \{p_i\}_{i \in I}$. The index set I controls the size of the point cloud as well as the amount of overlap between the sampling windows. Examples of the SW transformation can be seen in Figure **??**.

2.2. Dimension reduction

The topological construction we use in the following steps suffers from both computational and accuracy issues in high dimensions. Therefore, dimension reduction is a key step in our pipeline. There are numerous powerful and well-studied methods for performing dimension reduction (e.g., PCA, MDS, Isomap), and one has the freedom to choose any of them here. We chose to use the Uniform Manifold Approximation and Projection (UMAP) algorithm [**?**]. Briefly, UMAP is a manifold-learning algorithm, whose main objective is to find a low-dimensional graph-representation that preserves the global (topological) structure of the data. The main reasons for this choice are that: (a) The assumptions on the structure underlying the data are quite generic (a Riemannian manifold), supporting the intricate shapes generated by the SW transformation; (b) Preserving the global structure, implies that when we extract topological information from the data we do not lose much information.

2.3. Simplicial filtration

Simplicial complexes. Simplicial complexes are high-dimensional generalization of graphs, where in addition to vertices and edges we also include triangles, tetrahedra and higher dimensional faces. Taking the vertices to be the point cloud $\mathcal{P}_d$ (the output of the previous step), simplicial complexes serve as a discrete approximation (possibly a triangulation) of the underlying space, from which we can extract the desired topological information.

Formally, let S be a set, and X be a collection of nonempty finite subsets of S. We say that X is an *abstract simplicial complex* if it satisfies the following property,

$$A \in X, \text{ and } B \subset A \Longrightarrow B \in X.$$

Each element $A \in X$ is called a *simplex*, whose dimension is $|A|-1$. In particular, $|A| = 1$ is a vertex, $|A| = 2$ is an edge, etc.

The alpha complex. There are various ways to generate simplicial complexes from point clouds (e.g., the Čech and Vietoris-Rips complexes [**?**]), and this is also an algorithm-design choice. Here, we chose to use the alpha complex, which is constructed as follows.

Let $\mathcal{P}$ be a finite point cloud in $\mathbb{R}^d$. For each $p \in \mathcal{P}$ we define its *Voronoi cell* and *truncated Voronoi cell* as,

$$V(p) := \{x \in \mathbb{R}^d : \|x - p\| \leq \left\|x - p'\right\|, \ \forall p' \in \mathcal{P}\},$$
$$V_r(p) := V(p) \cap B_r(p),$$

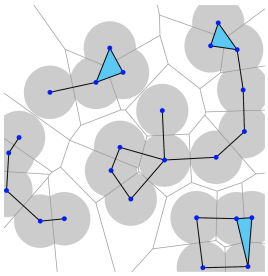

Fig. 2: Alpha complex for a 2D point cloud with given radius. The grey lines are the boundaries of the Voronoi cells $V(p)$, and the shaded regions are the truncated Voronoi cells $V_r(p)$.

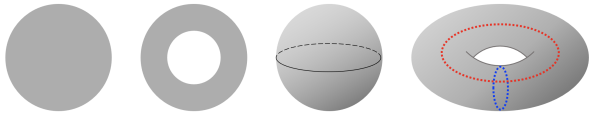

Fig. 3: Homology. All shapes here have a single connected component (H_0). The ring has a single hole (H_1), the sphere has a single "air pocket" (H_2), and the torus has two holes and one air pocket.

where $B_r(p)$ is the ball of radius r around p. The alpha complex $\mathcal{A}_r(\mathcal{P})$ is the *nerve* of the truncated Voronoi cells, i.e.,

$$\{p_1, \ldots, p_k\} \in \mathcal{A}_r(\mathcal{P}) \Longleftrightarrow \bigcap_{i=1}^{k} V_r(p_i) \neq \emptyset.$$

See Figure **??** for an example. Note that when $r = \infty$ we get that $\mathcal{A}_\infty(\mathcal{P})$ is the well-known Delaunay triangulation. One important property of the alpha complex (via the Nerve Lemma [**?**]), is that $\mathcal{A}_r(\mathcal{P})$ has the same homology as the union of balls $\bigcup_p B_r(p)$.

2.4. Persistent homology

Persistent homology is the main tool used in TDA which allows us to extract robust topological information from filtrations of spaces. In this section we provide a brief and intuitive explanation to what persistent homology is, and refer the reader to [**?**, **?**] for details.

Homology. Briefly, homology is an algebraic-topological structure that captures information about the shape of topological spaces and about mappings between spaces. If X is a topological space (e.g., a simplicial complex), we attach to it a sequence of vector spaces (or groups in the more general formulation) denoted $H_0(X), H_1(X), H_2(X)$, etc. The basis elements of $H_0(X)$ correspond to the *connected components* of X, $H_1(X)$ – to *loops surrounding holes* in X, $H_2(X)$ – to *closed surfaces enclosing "air pockets"* in X. Generally, $H_k(X)$ represents information about "k-dimensional cycles", which can be thought of as k-dimensional surfaces that are empty from within. See examples in Figure **??**.

In the context of data analysis, the power of the homology functional comes from the fact that it is independent of the coordinate system chosen, and is robust to continuous deformations. On the other hand, since the nature of homology is discrete (holes can either exist or not), it can be sensitive to noisy observations. One of the goals of persistent homology is to overcome this sensitivity.

Persistent homology. The goal of persistent homology is to analyze a *filtration* (increasing sequence) of spaces rather than a single

space, and to provide information about the "evolution" of k-cycles throughout the process. Suppose that we have a filtration $\mathcal{F} = \{X_t\}$ (i.e., $X_s \subset X_t$ for $s < t$). Considering the homology $H_k(X_t)$, as t increases, k-cycles (holes) may form at various times, and later fill in. The k-th persistent homology, $\mathrm{PH}_k(\mathcal{F})$, tracks this birth-death process. The information provided by $\mathrm{PH}_k(\mathcal{F})$ is often summarized by a *persistence diagram*, $\mathrm{dgm}_k(\mathcal{F})$, which is a collection of points in the plane, where the x-axis represents the birth time of a cycle, and the y-axis represents its death time. See Figure **??** for examples.

We apply persistent homology to the filtration of alpha complexes generated from the lower-dimensional point cloud $\mathcal{P}_d$. Here, the time parameter of the filtration is represented by the radius r. In other words, instead of choosing a fixed radius r and analyzing the homology of $\mathcal{A}_r(\mathcal{P})$, we consider the entire range of radii $[0, \infty)$, and analyze the persistent homology of the filtration $\{\mathcal{A}_r(\mathcal{P})\}_{r=0}^{\infty}$.

2.5. Extracting topological features

The complete pipeline presented in Sections **??**-**??** is

$$(x_n) \xrightarrow{(1)} \mathcal{P}_D \xrightarrow{(2)} \mathcal{P}_d \xrightarrow{(3)} \{\mathcal{A}_r(\mathcal{P}_d)\} \xrightarrow{(4)} \mathrm{dgm}_k(\{\mathcal{A}_r(\mathcal{P}_d)\}),$$

where the steps are: (1) SW transformation, (2) UMAP dimension reduction (can be replaced with other algorithms), (3) alpha complex filtration (can be replaced with other filtrations), and (4) persistent homology computation. See examples in Figure **??**. In this section we discuss the numerical values that we extract from the output persistence diagrams, and the motivation for using them.

The connection between temporal signals and their corresponding persistent homology was studied in the past from both theoretical and applied perspectives. The most intuitive connection is between periodic signals and PH_1, since periodicity in the signal results in loops formed in the point cloud. This connection was studied rigorously in [**?**], and has been used in various applications [**?**, **?**, **?**, **?**]. More recently, it was shown in [**?**] that quasi-periodic signals generate 'torus-like' point clouds, where the torus dimension depends on the number of independent frequencies forming the signals. This establishes a link to PH_k for higher values of k (a d-dimensional torus has cycles in dimension $k = 0, \ldots, d$). In addition to the above results, while experimenting with alarm sound signals, we discovered that signals with continuously shifting frequencies (e.g., a chirp, FM signals), often have distinctive features in PH_2 ("air pockets"), see for example the second row in Figure **??**.

Next, we provide formulae for the numerical features extracted from the persistence diagrams computed for a single time series (x_n). Note that the quantities described next were defined in a heuristic way. Nevertheless, we will try to provide some intuition behind their definition. Our features make use of the following notation. Recall that each diagram dgm_k is a collection of $(\mathrm{death}, \mathrm{birth})$ pairs, representing the lifetime of each k-cycle in PH_k. We define $\mathcal{L}_k = \{L_{k,1}, L_{k,2}, \ldots\}$ to be the collection of lifetime lengths (i.e., $\mathrm{death} - \mathrm{birth}$) of all cycles in PH_k. We assume that the list is sorted in a descending order, so that $L_{k,1}$ represents the longest lifetime. The first type of quantities we define aims to capture the signal properties discussed above. To this end, we define

$$\mathrm{PS} = 1 - \frac{L_{1,2}}{L_{1,1}}, \quad \mathrm{QPS} = L_{1,2} \cdot L_{2,1}, \quad \text{and} \quad \mathrm{FSS} = \frac{L_{2,1} \cdot L_{2,2}}{L_{1,1}}.$$

The Periodicity Score (PS) checks how close the signal is to being periodic, i.e., having a single dominant 1-cycle. The Quasi Periodicity Score (QPS) tests for quasi-periodicity. This score is high if there are at least two significant 1-cycles and a significant 2-cycle,

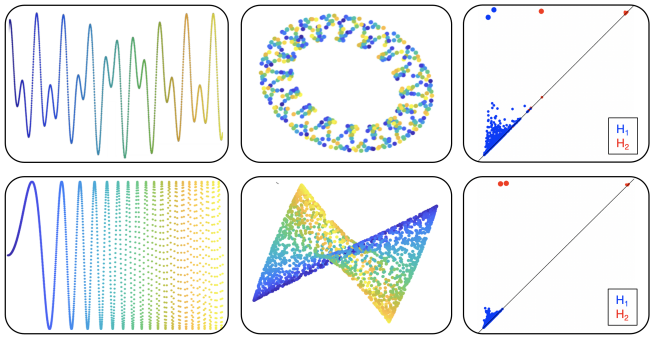

Fig. 4: From signal to persistence diagrams. Each row presents a discrete-time signal, its corresponding SW transformation in $\mathbb{R}^3$, and the persistence diagrams $\mathrm{dgm}_1, \mathrm{dgm}_2$. Top row: a quasi-period signal with two independent periods. The point cloud generated has the shape of a torus. Indeed, its persistence diagrams have two persistent features (far from the diagonal) in dgm_1, and one in dgm_2. Bottom row: a chirp signal. Here the point cloud seems to generate two air-pockets, manifested by the two persistent features in dgm_2.

an indication for a torus-like shape (see Figure **??**). The Frequency Shift Score (FSS) checks for the appearance of a pair of dominant 2-cycles, which in our study were shown to be indicative of frequency shifts (Figure **??**). The second type of features we extract aims to provide statistical information about the lifetime distributions of the cycles. For each $k \in \{0, \ldots, 4\}$ we compute the following:

- Top longest lifetimes: $L_{k,i}$, $i = 1, \ldots, 5$.
- Number of α-long lifetimes: $N_{k,\alpha} = \#(L_{k,i} > \alpha)$. The value of α is chosen empirically to distinguish between meaningful features and "noise" in the diagram.
- Mean and variance: $m_k = \mathrm{mean}(\mathcal{L}_k)$, $\sigma_k^2 = \mathrm{var}(\mathcal{L}_k)$.
- Normalized longest lifetime: $L_{k,1}/|\mathcal{L}_k|$, and $L_{k,1}/m_k$.

In addition, we compute values that measure the interplay between cycles in different dimensions. For all $k_1 < k_2 \in \{0, 1, 2\}$ we take the following:

- Ratio of means: m_{k_1}/m_{k_2}.
- Ratio of α-long cycles $N_{k_1,\alpha}/N_{k_2,\alpha}$.
- Products of top longest lifetimes: $L_{k_1,i} \cdot L_{k_2,i}$, $i = 1, \ldots, 6$.
- The product: $L_{k_1,i} \cdot (L_{k_2,i} - L_{k_2,i+1})$, $i = 1, \ldots, 6$.

3. EXPERIMENTAL RESULTS

The dataset we used to demonstrate the power of our framework is the UrbanSound8K[**?**]. This dataset contains 8,732 sound excerpts from 10 different classes, 929 of them are labeled as alarm sounds. The excerpts are at most 4 seconds long, and the sampling rate is 44,100[Hz]. In order to speed up the topological computations, and since we observed little if any performance loss, we down-sampled the signals to 8,820[Hz]. Since alarm sounds form only one class out of ten in this dataset, we used a bagging method in the training stage (balanced batches with 32 alarm and non-alarm sounds). The excerpts were sampled with replacement. The model was trained incrementally on the balanced batches, and the procedure was repeated until every excerpt in the training set was used.

3.1. Additional implementation details

Classical DSP features. In addition to the novel topological features presented in Section **??**, we also incorporate classical DSP quantities in our classification task. We use similar features as in [**?**], where they were shown to be powerful, specifically in the context of alarm sound detection: (1) Pitch (using the YIN algorithm [**?**]); (2) Short time energy; (3) Zero crossing rate; (4) 13 Mel-frequency cepstral coefficients; (5) Spectral flux; (6) Spectral roll-off; (7) Spectral centroid; (8) Spectral flatness. For (2)-(8) we used 20[ms] windows, and took the maximum, minimum, average, median, and standard deviation over all windows. In addition, as in [**?**], we computed the Discrete Wavelet Transform (DWT) with 10 decomposition levels. We calculated the energy and waveform length for each level, and took their variance across the levels.

Feature selection. We presented above a long list of features (topological and DSP) that are potentially useful for our signal classification task. In order to decide which features should be used in the training stage, we use the *ReliefF* algorithm [**?**, **?**] to assign weights to each feature, measuring its relative contribution to the classification problem at hand. Briefly, this algorithms iteratively ranks features used by a binary classifier, according to the difference in values between data points and their nearest hit (same-class instance) and nearest miss (different-class instance). The complete feature ranking is available online (see github link below). We note that among the 15 highest ranked features, 9 are topological features including the PS and QPS scores, which confirms our intuition that the shape of the point cloud serves as a significant statistic. In addition, among the top 20 highest ranked topological features, we note that four are derived from PH_2, and one from PH_3, indicating that homology in degrees higher than 1 also plays a significant role.

Classifier. Current state-of-the-art results for alarm sound detection, are achieved using powerful machine learning algorithms (e.g. DNNs). Since our goal here is to focus on new powerful *features*, we intentionally chose to use a classical algorithm, and to highlight the contribution of the features rather than the algorithm. We use here the AdaBoost classifier [**?**] with shallow decision trees, and with one proprietary modification: As part of its training stage, the AdaBoost algorithm trains many decision trees separately, and then combines them together. Since the number of features we are generating is very large, instead of using all of them to train all the decision trees, we randomly sample a small subset of features for each tree. The way we generate this sample is such that the probability to choose the i-th feature is proportional to e^{w_i}, were w_i is the weight of i-th feature as calculated by the ReliefF algorithm. This way, we guarantee a bias towards the more informative features. The sample size we used was 4 (from an ensemble of 120 features).

3.2. Results

For each of the sets (train, validation, test), we extracted both the topological and DSP features as described in Sections **??** and **??**, respectively. We applied the ReliefF algorithm to the train set in order to rank the contribution of all features. In the classifier we used all $N_{\mathrm{TOP}} = 80$ topological features that obtained a positive ReliefF weight, and added the top $N_{\mathrm{DSP}} = 40$ DSP features. For the SW transformation we used $D = 70$, and for UMAP algorithm we took $d = 4$. The choice of the step-size τ for the SW transformation was made heuristically by searching for the value that maximizes the average QPS score for the train set. The tuning of all these parameters was done in using k-fold cross validation. The results are presented in first row of Table **??**(a).

Features	Accuracy
TOP & DSP	98.8%
TOP1 & DSP	98.1%
TOP	94.6%
DSP	71.0%

(a)

Method	Accuracy
Zhang et al [**?**]	96.4%
Garg et al [**?**]	96.7%
Li et al [**?**]	98.7%
Current work	98.8%

(b)

Table 1: (a) The accuracy results of the combined framework, compared to topological-only and DSP-only features. By TOP1 we refer to features derived from dgm_0 and dgm_1 only. (b) Comparing our results with the most recent state of the art.

To assess the contribution of topological features to the final results, we ran a small ablation study. We ran the algorithm once with 40 topological features only (no DSP), and once with 40 DSP features only. We also included a run with topological features only in dimensions 0 and 1, to assess the contribution of the higher dimensional features. The results in Table **??**(a) highlight the remarkable performance of the topological features, and also show that topological features in dimensions $2-4$ have notable contribution.

Reported accuracy results in previous work [**?**, **?**, **?**, **?**, **?**, **?**, **?**] (albeit, for different datasets) range between 85%-98.49%, so the accuracy we achieved by including topological features is higher. With respect to the UrbanSound8K dataset, we present the state of the art results in Table **??**(b). Note that the results reported in [**?**, **?**, **?**] are for the full classification problem (with 10 classes). Therefore, in order to compare to our setting (binary classification), we computed the marginal accuracy of their results, with respect to alarm sounds vs. other classes. He as well, our results have the highest accuracy. We note that all other methods presented in table **??**(b) are based on deep CNNs. Intuitively, the higher accuracy of our classifier stems from the use of global topological features, which are relatively robust to small perturbations, and make it less susceptible to low SNR.

4. CONCLUSION

We presented a new approach to alarm sound detection using novel topological features, providing state-of-the-art results, while using a relatively simple classifier. The framework we develop here is quite generic and can be applied in other signal classification problems, with minor adjustments. With respect to the UrbanSound8K dataset, it would be interesting to test the capabilities of the topological framework in the full classification problem (i.e., all 10 classes). One important direction of future research is to better understand the topological features that were found to be the most prominent, and to use this knowledge to devise a systematic way of extracting features from persistence diagrams. This should hopefully lead to using the framework we propose here in many other settings.
Code is available at `https://github.com/tofi98/alarms`.

Acknowledgement The authors would like to thank Yair Moshe for suggesting alarm sounds as an interesting test case for our framework. We also wish to thank Ori Bryt, Nimrod Peleg and Ronen Talmon for their support and helpful advice.

5. REFERENCES

[1] B. Fatimah, A. Preethi, V. Hrushikesh, A. Singh B., and H. R. Kotion, "An automatic siren detection algorithm using fourier decomposition method and mfcc," in *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 2020, pp. 1–6.

[2] D. Rane, P. Shirodkar, T. Panigrahi, and S. Mini, "Detection of ambulance siren in traffic," in *2019 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET)*, 2019, pp. 401–405.

[3] D. Carmel, A. Yeshurun, and Y. Moshe, "Detection of alarm sounds in noisy environments," in *2017 25th European Signal Processing Conference (EUSIPCO)*, 2017, pp. 1839–1843.

[4] F. Meucci, L. Pierucci, E. Del Re, L. Lastrucci, and P. Desii, "A real-time siren detector to improve safety of guide in traffic environment," in *2008 16th European Signal Processing Conference*, 2008, pp. 1–5.

[5] Z. Chi, Y. Li, and C. Chen, "Deep convolutional neural network combined with concatenated spectrogram for environmental sound classification," in *2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT)*, 2019, pp. 251–254.

[6] S. Garg, T. Sehga, A. Jain, Y. Garg, P. Nagrath, and R. Jain, "Urban sound classification using convolutional neural network model," *IOP Conference Series: Materials Science and Engineering*, vol. 1099, no. 1, p. 012001, 2021.

[7] Z. Zhang, S. Xu, S. Cao, and S. Zhang, "Deep convolutional neural network with mixup for environmental sound classification," in *Pattern Recognition and Computer Vision*, J.-H. Lai, C.-L. Liu, X. Chen, J. Zhou, T. Tan, N. Zheng, and H. Zha, Eds. Cham: Springer International Publishing, 2018, pp. 356–367.

[8] G. Carlsson, "Topology and data," *Bulletin of the American Mathematical Society*, vol. 46, no. 2, pp. 255–308, 2009.

[9] A. Zomorodian, "Topological data analysis," *Advances in Applied and Computational Topology*, vol. 70, pp. 1–39, 2007.

[10] S. Bhattacharya, R. Ghrist, and V. Kumar, "Persistent homology for path planning in uncertain environments," *IEEE Transactions on Robotics*, vol. 31, no. 3, pp. 578–590, 2015.

[11] H. Kannan, E. Saucan, I. Roy, and A. Samal, "Persistent homology of unweighted complex networks via discrete morse theory," *Scientific reports*, vol. 9, no. 1, pp. 1–18, 2019.

[12] Y. Wang, H. Ombao, and M. K. Chung, "Statistical persistent homology of brain signals," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 1125–1129.

[13] L. Aloni, O. Bobrowski, and R. Talmon, "Joint geometric and topological analysis of hierarchical datasets," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2021, pp. 478–493.

[14] J. A. Perea and J. Harer, "Sliding windows and persistence: An application of topological methods to signal analysis," *Foundations of Computational Mathematics*, 2014.

[15] H. Gakhar and J. A. Perea, "Sliding window persistence of quasiperiodic functions," *arXiv preprint arXiv:2103.04540*, 2021.

[16] J. A. Perea, A. Deckard, S. B. Haase, and J. Harer, "Sw1pers: Sliding windows and 1-persistence scoring; discovering periodicity in gene expression time series data," *BMC bioinformatics*, vol. 16, no. 1, pp. 1–12, 2015.

[17] S. Emrani, T. Gentimis, and H. Krim, "Persistent homology of delay embeddings and its application to wheeze detection," *IEEE Signal Processing Letters*, vol. 21, no. 4, pp. 459–463, 2014.

[18] S. Emrani, H. Chintakunta, and H. Krim, "Real time detection of harmonic structure: A case for topological signal analysis," in *2014 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2014, pp. 3445–3449.

[19] C. J. Tralie and M. Berger, "Topological eulerian synthesis of slow motion periodic videos," in *25th IEEE international conference on image processing (ICIP)*. IEEE, 2018, pp. 3573–3577.

[20] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," *arXiv preprint arXiv:1802.03426*, 2018.

[21] K. Borsuk, "On the imbedding of systems of compacta in simplicial complexes," *Fundamenta Mathematicae*, vol. 35, no. 1, pp. 217–234, 1948.

[22] A. Hatcher, *Algebraic topology*. Cambridge: Cambridge University Press, 2002.

[23] J. Salamon, C. Jacoby, , and J. P. Bello, "A dataset and taxonomy for urban sound research," *22nd ACM International Conference on Multimedia(ACM-MM'14)*, 2014.

[24] A. de Cheveigné and H. Kawahara, "Yin, a fundamental frequency estimator for speech and music," *The Journal of the Acoustical Society of America*, vol. 111, no. 4, pp. 1917–1930, 2002.

[25] K. Kira, L. A. Rendell *et al.*, "The feature selection problem: Traditional methods and a new algorithm," in *Aaai*, vol. 2, no. 1992a, 1992, pp. 129–134.

[26] I. Kononenko, E. Šimec, and M. Robnik-Šikonja, "Overcoming the myopia of inductive learning algorithms with relieff," *Applied Intelligence*, vol. 7, no. 1, pp. 39–55, 1997.

[27] R. E. Schapire, *Explaining AdaBoost*. Springer Berlin Heidelberg, 2013, pp. 37–52.

[28] L. Marchegiani and P. Newman, "Listening for sirens: Locating and classifying acoustic alarms in city scenes," *CoRR*, vol. abs/1810.04989, 2018.

[29] V.-T. Tran and W.-H. Tsai, "Acoustic-based emergency vehicle detection using convolutional neural networks," *IEEE Access*, vol. 8, pp. 75 702–75 713, 2020.

[30] J.-J. Liaw, W.-S. Wang, H.-C. Chu, M.-S. Huang, and C.-P. Lu, "Recognition of the ambulance siren sound in taiwan by the longest common subsequence," in *2013 IEEE International Conference on Systems, Man, and Cybernetics*, 2013, pp. 3825–3828.

[31] J. Schröder, S. Goetze, V. Grützmacher, and J. Anemüller, "Automatic acoustic siren detection in traffic noise by part-based models," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 493–497.