Anomalous Sound Detection For Road Surveillance Based On Graph Signal Processing

Zied Mnasri, Jhony Heriberto Giraldo Zuluaga, Thierry Bouwmans

Anomalous Sound Detection For Road Surveillance Based On Graph Signal Processing

Zied Mnasri [a,d], Jhony H. Giraldo [b], Thierry Bouwmans [c]
[a] University of Bradford, UK, [b] LTCI, Télécom Paris, Institut Polytechnique de Paris, France,
[c] Université La Rochelle, France, [d] University of Tunis El Manar, Tunisia
z.mnasri@bradford.ac.uk, jhony.giraldo@telecom-paris.fr, thierry.bouwmans@univ-lr.fr

***Abstract*—Recently, Anomalous Sound Detection (ASD) has emerged as a promising method for road surveillance. However, since the ratio of anomalous events is generally too small, anomaly detection in general, and ASD in particular, are mainly treated as one-class classification problems. Besides, a common problem in road surveillance is the background noise, which makes it difficult to train effective models based on normal sounds only. Therefore, this work aims to experiment with the use of graph signal processing (GSP) to improve ASD performance. Thus, we propose a Graph-based One-Class SVM technique (GOC-SVM) where the features extracted from audio signals are firstly embedded on graphs, and then filtered through a graph filterbank, before computing their joint Fourier transform magnitude. Subsequently, they are fed into a one-class SVM classifier trained on normal data only. Evaluation results show a threefold advantage of using graph embedding and filtering for ASD: (a) improving the anomaly detection results in comparison to plain features, (b) outperforming the classical OC-SVM baseline, (c) enhancing the performance of the proposed semi-supervised GOC-SVM, so as to reach a comparable level of performance of the fully-supervised binary classification SVM, yielding 0.91 of Area-under-the-curve (AUC), 98% of overall accuracy, 99% and 88% of F1 score for normal and anomalous classes, respectively. Such a performance proves the potential of using GSP to solve the ASD problem in road traffic monitoring.**

***Index Terms*—Sound event detection, anomaly detection, audio surveillance, graph signal processing, joint Fourier transform, one-class SVM.**

I. Introduction

A. Anomalous sound detection (ASD)

Audio event detection offers novel options to improve surveillance techniques in general, and road traffic monitoring in particular. Even though video road surveillance has attained a considerable accuracy [1], using audio data can still outperform video in certain circumstances, mainly due to its insensitivity towards luminosity conditions and its omnidirection [2]. However, some major problems of anomaly detection in general and anomalous sound detection in particular are still to be resolved. First, the scarcity of anomalous data, in comparison to normal data, makes it difficult to find enough anomalous samples for training [3]. Secondly, anomalous event characterization is related to the type of the event, and partly to the acoustic scene, which makes it difficult to find a standard feature set to characterize anomalous audio events [4]. Thirdly, audio event signals in real environments are generally mixed with background noise and/or other events, which may be confusing while detecting the proper event [5].

B. Graph signal processing (GSP) and its application to audio signals

The aforementioned issues make it difficult to reach high performances using semi/weakly-supervised learning methods for ASD. Therefore, novel solutions for ASD are still to be proposed, such as the use of graph signal processing (GSP). Recently, graphs have gained increasing interest for their ability to model several types of data and complex interactions among them, *e.g.*, images and their pixels, users on a social network and their connections, sensors and their measurements. In all these cases, emitters and their data/interactions can be modeled as nodes and edges, respectively [6]. Considering the recent applications of GSP to resolve anomaly/outlier/novelty detection problems in several fields, such as social networks, computer traffic, financial transactions [7], and also the proved efficiency of GSP to provide an alternative signal description, mainly for image and video [8], this work aims at prospecting the potential of GSP to resolve the ASD problem.

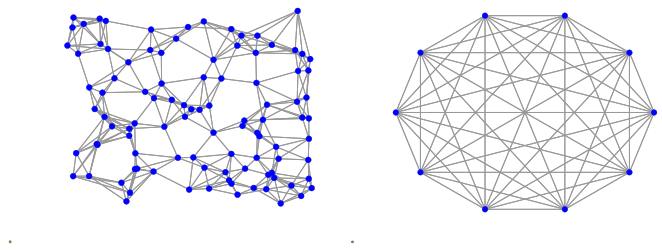
C. Scope and contribution

Therefore, this paper describes a novel and experimental work aiming at detecting anomalous sound events using GSP combined with semi-supervised one-class support vector machines (OC-SVM). Thus, the novelty consists in experimenting the application of GSP for signal embedding on graphs, to extract significant features that will be used to train an OC-SVM model on *normal* data only. To fulfill this goal, several graph topologies are tested to achieve signal embeddings, such as *sensor*, *path*, *ring*, *fully-connected (FC)* and *nearest neighbors (NN)* graphs. Finally, benchmarking is carried out using an ablation study scheme, based on the proposed workflow.

II. Preliminaries

A. Anomaly detection

It is first necessary to define the notion of anomaly/novelty/outlier. In [9], this is characterized using (a) its scarcity, as *anomalous/novel/outlier* events occur less frequently than *normal* events; (b) its characteristics, as anomalous/novel/outlier events should have different



(a) Sensor (b) Fully-connected

Fig. 1: Examples of graph structures used for embedding the audio clips: For each graph structure, the number of nodes is equal to the number of features extracted from the audio chunk

characteristics than *normal* events; (c) its meaning, as such events should carry a specific and a different meaning than normal events. Therefore, in most cases, conventional classification techniques are not quite efficient to detect anomalies/outliers, as anomalous patterns are too few, and also too heterogeneous to form a separate and unique cluster.

B. Graph Signal

Consider a dataset with N elements, for which some relational information about its data elements is known, such as the examples given in Figures 1a and 1b. This information can be represented by a graph $G = (\mathcal{V}, \mathbf{A})$, where $\mathcal{V} = \{v_0, \ldots, v_{N-1}\}$ is the set of nodes and $\mathbf{A}$ is the weighted adjacency matrix of the graph. Each dataset element corresponds to a node v_n, whereas each weight $A_{n,m}$ of an edge directed from v_m to v_n reflects the degree of relation of the n^{th} node to the m^{th} one. Since data elements can be related to each other differently, in general, G is a directed, weighted graph. The set of indices of nodes connected to v_n is called the neighborhood of v_n and denoted by $\mathcal{N}_n = \{m \mid A_{n,m} \neq 0\}$ [10].

Assuming, without a loss of generality, that dataset elements are complex scalars, we define a graph signal as a map from the set of nodes $\mathcal{V}$ into the set of complex numbers $\mathbb{C}$:

$$\begin{array}{rccc} s & : & \mathcal{V} & \rightarrow & \mathbb{C}, \\ & & v_n & \rightarrow & s_n. \end{array} \tag{1}$$

Notice that each signal is isomorphic to a complex-valued vector with N elements. Hence, for simplicity of discussion, we write graph signals as vectors $\mathbf{s} = (s_0, s_1, \ldots, s_{N-1})^T$. Each element s_n is indexed by node v_n of a given representation graph $G = (\mathcal{V}, \mathbf{A})$, as defined by (1). The space $\mathcal{S}$ of graph signals in (1) is then identical to $\mathbb{C}^N$.

III. Related work

The two main strategies for ASD are static and dynamic modeling. In the first case, signals are embedded in a representation space and anomalies are detected either by distance in the latent representation or by reconstruction error [5]. In the second case, the temporal evolution is evaluated against models of "background" [2]. Nevertheless, the two strategies may be combined, using either fully-, semi-, or un-supervised learning:

1. *Fully-supervised learning* – Several classifiers based on recurrent/convolutional neural networks have been recently proposed for ASD, both statically and dynamically, *e.g.*, [11], [12]. For instance, in [4], a novel method for detecting road accidents through audio stream analysis is proposed.
2. *Semi-supervised learning*– This method, mostly based on one-class support vector machines (OC-SVM), is a classical static anomaly detection tool, that has been applied to anomalous sound event detection in [3], [4] . Recently, in [13], the authors have proposed an ensemble one-class SVM parallel to an MLP network to calculate the anomaly score for audio events. Since OC-SVM is used throughout this work, a comprehensive explanation of the OC-SVM formulation is provided in [14].
3. *Unsupervised learning* – Several works leveraging deep/variational autoencoders, in some cases using dynamic modeling, have been recently proposed for anomalous sound event detection, *e.g.*, [15]. In particular, the proposal by Wei *et al.* [16] at DCASE 2020 challenge-Task 2 is based on a reconstruction autoencoder to calculate the anomaly score through metric learning. Therefore, different types of autoencoders are tested, such as deep autoencoders, variational autoencoders, etc.

Regarding GSP, up to our knowledge, it has not been applied to ASD so far. However, it has efficiently been utilized for anomaly detection in several areas, such as image and video processing [10]. Besides, graph embedding has been proven to provide a feature description that helps improve classification results in several problems, such as social networks, computer traffic, and financial transactions [7].

IV. Methods

In this work, a novel GSP-based method for ASD is proposed. The method is articulated on three main components, namely feature extraction, graph embedding, and OC-SVM classification. Moreover, it is compared to some previous and novel benchmarking methods, also built on some of the aforementioned components.

A. Proposed method: Graph signal embedding with One-class SVM classification (GOC-SVM)

The main and novel idea behind this proposal is the use of graphs to embed the input feature vectors before performing one-class classification using OC-SVM. The reason motivating such a choice is the noticed lower discrimination power of the plain audio features, *i.e.*, without graph embedding, in the context of audio traffic data, where all signals, whether *normal* or *anomalous* are corrupted by the background noise [5]. The workflow of the proposed method is based on three main components, namely *Feature extraction*, *Graph signal processing* and *One-class classification*, as depicted in path (b) in Figure 2):

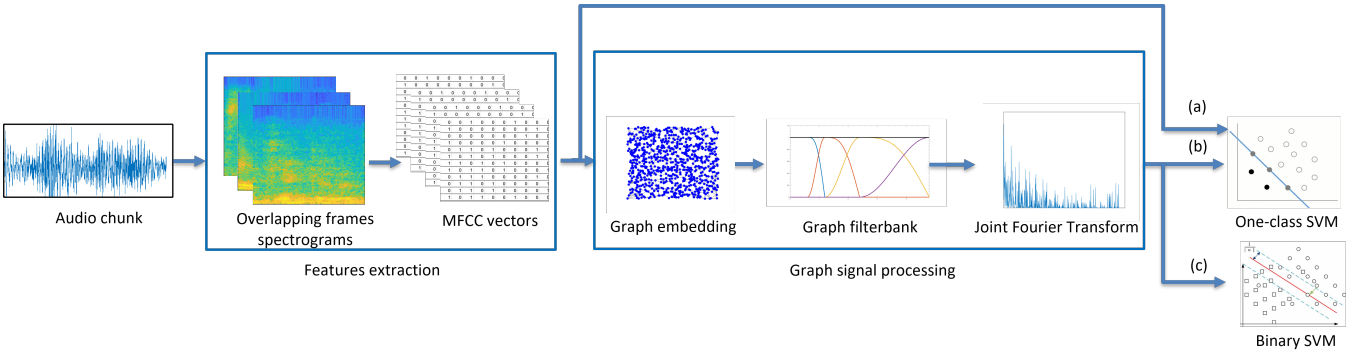


Fig. 2: Workflow of the presented methods: (a) OC-SVM without graph signal processing (Benchmark 1), (b) Graph signal processing with One-Class SVM classification (GOC-SVM) (Proposed), (c) Graph signal processing with Binary SVM classification (GBin-SVM) (Benchmark 2-Proposed)

1. *Feature extraction*- The basic features used in this method are the Mel-Frequency Cepstral Coefficients (MFCC). It should be noted that they have been selected among other types of audio features for their proven performance in ASD [4], [12], [15]. To extract the MFCC features, the audio chunk is processed as follows:

1.1. Each audio clip of the database (of approximately 1 minute) is segmented into short chunks of 1 second so that each chunk contains at most one *anomalous* event. Hence each chunk is labeled as *normal* or *anomalous*.

1.2. For each chunk, the Mel-log spectrogram is calculated using overlapping frames of 1024 points with 50% overlap.

1.3. For each frame, a vector of 14 MFCC coefficients representing MFCC0 $= \log(energy)$ and MFCC(1)-MFCC(13), are extracted from the Mel-log spectrogram. Thus, for each chunk, a matrix of dimension $N_f \times 14$ is obtained, where N_f is the number of frames in the chunk. Then, this matrix is flattened into a 1-D supervector.

2. *Graph signal processing*- This is the core component of the proposed method, where graph structures are utilized to embed the extracted features:

2.1. For each chunk, a graph is constructed, and the MFCC supervector is embedded into it. It should be noted that, for each chunk, the MFCC's supervector is embedded in the same type of graph (like Path, Ring, etc) used for all other chunks. Thus, all chunks are embedded in the same graph structure, using the same weight and node parameters.

2.2. The embedded MFCC supervector is filtered through a *Simple Tight Frame filterbank*. It is worth noting that this step is optional, and is useful to assess the effect of filtering on graph embedding.

2.3. For each filtered/unfiltered graph-embedded MFCC supervector, the Joint Fourier Transform (JFT) is computed, and its magnitude is returned. At this level, JFT is computed on the time-graph scale and is analogous to the classical STFT in the time-frequency domain. However, JFT is useful to extract the eigenvalues of the graph Laplacian and thus is intended to discover some latent features [17].

3. *One-class classification*- This component is used to perform semi-supervised classification, as follows:

3.1. One-class SVM algorithm is used to train the obtained JFT magnitude values of the graph-embedded MFCC vectors. Note that the One-class SVM is trained only using the *normal* data.

3.2. Once One-class SVM training is finished, testing is applied on a subset containing both *normal* and *anomalous data*.

Several types of graph topologies were experimented, either using a temporal structure, such as the *Path* and the *Ring* graphs, or not, *e.g.*, the *Sensor* and the *Fully-connected* graphs [17]. Samples of such graphs are illustrated in Figure 1.

B. Benchmarking methods

To evaluate the results and the effectiveness of the proposed method, two benchmarking methods are tested, using an ablation-study scheme, where one component of the proposed method is deleted or modified at a time. Thus, this strategy produced the following benchmarks:

(a) benchmark 1- One-class SVM (OC-SVM): This is the basic benchmark, already used in [3], [4], [13], where only the feature extraction component and the OC-SVM classifier are used, bypassing the GSP module (cf. path (a) in Figure 2). Thus, the aim of this benchmark is to assess the effectiveness of using/not using graph embedding in semi-supervised ASD.

(b) benchmark 2- Graph-based binary SVM (GBin-SVM): This is a novel benchmarking method that includes all the proposed components, *i.e.*, feature extraction, graph signal processing, and classification. However, it uses binary SVM instead of OC-SVM for classification (cf. path (c) in Figure 2). Hence, this second benchmark aims at assessing the contribution of using GSP to enhance semi-supervised classification results, yielded by the proposed

GOC-SVM, in comparison to a fully-supervised classifier, *i.e.*, Binary SVM.

V. Experiments and results

A. Audio materials

To perform this work, a database of audio surveillance on roads had to be selected. Among the available databases, MIVIA [4] has been chosen for the following reasons: a) Recordings were realized in a real road environment covering the city center, highways, and country roads; b) the total duration of the database is approximately one hour, divided in 57 audio clips, which provides a sufficient amount of data; c) the recorded sounds are labeled manually, indicating the audio event and its onset and offset times; d) finally, and perhaps most importantly, all audio signals, whether considered normal or anomalous, contain similar levels of environmental background noise.

B. Experiments

The audio clips are segmented into chunks of 1 sec, so that for each chunk there is at most one *anomalous* event. Then, the proposed GOC-SVM and the benchmark methods, OC-SVM and GBin-SVM are applied as described in Section IV. The experimental protocol is conducted by making several choices regarding the following aspects:

1) Graph structures: Several graph structure types were utilized, namely *Sensor Network*, *Path*, *Ring*,*Fully-Connected (FC)* and *Nearest Neighbors (NN)* (cf. Figure 1). Each audio chunk is embedded in the same graph structure as the other ones and uses the same initial weight and node parameters. It should be noted that these graph structures have been selected for either their temporal structure, *e.g.*, *Path*, *Ring*, or their potential performance *e.g.*, *Sensor* and *FC*.

2) Graph embedding vs. No-graph using One-class SVM classification: One-class SVM is performed for both the plain MFCC coefficients (for the basic Benchmark 1, cf. path (a) in Figure 2) and the JFT-MFCC ones (for the proposed GOC-SVM and Benchmark 2 methods, cf. path (b), path (c) in Figure 2, respectively), using 4000 chunks for training (all *normal*) and 1000 for the test, containing *normal* and *anomalous* samples. Table II details the obtained results of *Area under the curve* (AUC), *overall accuracy*, and *F1 score* for *normal* and *anomalous* samples.

3) Graph embedding and filtering vs graph embedding only: One-class SVM is also performed on the JFT of the filtered graph-embedded MFCC features, to compare them with the JFT graph-embedded ones (without filtering). Filtering is achieved using a simple TF wavelet filter [17], applied on $N_{fb} = 6$ filterbank components and of order 6. However, it is worth noting that the filter type and its configuration have been set amongst several other ones, according to their final performances (cf. Table I and Table II).

4) OC-SVM vs. Binary SVM: Table I details the results of using one-class and binary-classification SVM. Both methods are tested using (a) *Unfiltered* JFT-MFCC features embedded on the Ring graph (for its highest performance amongst other types of graphs, cf. Table II), and (b) *Filtered* JFT-MFCC features embedded on the fully-connected (FC) graph (also selected for its highest performance, cf. Table II). Such an assessment is useful to understand the effect of using both graph embedding and filtering in enhancing the performance of the semi-supervised OC-SVM to make it as close as possible to a fully-supervised classifier, such as the Binary-classification SVM.

C. Results and discussion

Firstly, the results of Table II confirm the advantage of using both graph embeddings on the OC-SVM classification results, from either the overall metrics, *i.e.*, *AUC* and *Accuracy*, or the class-wise ones, *i.e.*, *F1* score for both classes, *normal* and *anomalous*. Such an improvement in comparison to the Benchmark 1 method, *i.e.*, OC-SVM, is especially noted when using a time-structured graph topology, such as the *Ring* graph.

Secondly, Table I confirms that applying a filterbank at the output of the graph remarkably improves the graph embedding performance for OC-SVM, *i.e.*, the proposed GOC-SVM, to make it comparable to binary classification SVM, *i.e.*, benchmark 2 (GBin-SVM). This is quite an interesting result since graph embedding and then filtering seem to provide a latent description that may be comparable to deep features, *i.e.*, descriptors that may increase the classification results even though they are not explicitly interpretable.

Thirdly, as mentioned in Table I, graph embedding and filtering allow the OC-SVM to reach much higher performances than using plain MFCC features, *i.e.*, with *No Graph*. This is proved by the fact that OC-SVM based on graph-embedded features (GOC-SVM) performs nearly as well as fully-supervised binary-classification SVM (GBin-SVM), even though the F1-score for the *anomalous* class needs to be improved. However, such a finding might be very promising to address one-class classification in particular, and weakly-supervised classification problems in general, through exploring the advantages of using graph embedding for timely-structured data, such as audio events.

Finally, and perhaps most importantly, the obtained results confirm that graph embedding may be a good alternative to deal with uncertainty in audio data. As already mentioned, all sounds in the dataset contain the same amount of background noise, which, at a certain level, would make it difficult to set separate clusters for anomalous and normal sounds. This problem might be addressed by embedding the features on an appropriate graph structure and choosing the right graph filter type.

VI. Conclusion

In this work, we tried to address the problem of Anomalous Sound Detection (ASD) using Graph Signal Processing (GSP), particularly through graph embedding (and optionally filtering) of the audio features, *i.e.*, MFCC coefficients, that are fed into a one-class SVM classifier. Results show, firstly a clear improvement in comparison to the ASD semi-supervised state-of-the-art method, *i.e.*, One-class SVM based on only

TABLE I: Classification results using (a) One-Class SVM (benchmark 1), (b) Graph-based One-Class SVM (GOC-SVM) (Proposed method) and (c) Graph-based Binary SVM (GBin-SVM) (benchmark 2-Proposed), for different signals of a clip duration of 1 sec), applied on (i) the original features (No Graph), (ii) features embedded on the best performing graph structure without filtering (Ring graph), (iii) features embedded on the best performing graph structure with filtering (FC graph), as highlighted in Table II

Classifier	**Graph use**	**Graph filter use**	**AUC**	**Acc**	**F1 (norm.)**	**F1 (anom.)**
OC-SVM (Benchmark 1)	No Graph	No filtering	0.56	0.23	0.21	0.25
Graph-based Binary SVM (GBin-SVM) (benchmark 2-Proposed)	Ring graph	No filtering	**0.90**	**0.98**	**0.99**	**0.89**
	FC graph	Simple TFW filterbank	**0.90**	**0.98**	**0.99**	**0.89**
Graph-based OC-SVM (GOC-SVM) (Proposed)	Ring graph	No filtering	0.66	0.42	0.49	0.31
	FC graph	Simple TFW filterbank	**0.91**	**0.97**	**0.98**	**0.88**

TABLE II: Classification results using the proposed Graph-based One-Class SVM (GOC-SVM) for different a clip duration of 1 sec, applied on (a) features embedded on a graph only, and (b) features embedded on a graph with Simple Tight Frame Wavelet (TFW) filterbank (Sensor Network, Path, Ring, Fully connected (FC) and Nearest Neighbors (NN)) (* The number of asteriscs indicate the performance rank of graph topologies)

Proposed GOC-SVM Method	**Graph type**	**AUC**	**Acc**	**F1 (normal)**	**F1 (anomalous)**
Using Graph embedding only	Sensor	0.61	0.34	0.37	0.30
	Path	0.59	0.29	0.31	0.27
	Ring***	**0.66**	**0.42**	**0.49**	**0.31**
	Fully-connected*	0.57	0.42	0.24	0.26
	Nearest neighbors**	0.66	0.41	0.48	0.33
Using Graph embedding with Simple TFW filterbank	Sensor	0.91	0.84	0.90	0.57
	Path	0.89	0.82	0.88	0.59
	Ring*	0.89	0.82	0.88	0.59
	Fully-connected***	**0.91**	**0.97**	**0.98**	**0.88**
	Nearest neighbors**	0.91	0.84	0.90	0.60

MFCC (without graph-embedding); and secondly a comparable performance with fully-supervised binary-classification SVM, in terms of objective metrics, such as AUC, accuracy and F1 scores. Such results imply that using GSP for feature embedding in the ASD framework helps increasing the clusters separability, and thus improving the discrimination power of the extracted features, even though the training is performed on *normal* data only. As an outlook, this work should be continued to understand why some specific graph topologies, like *Ring* and *Fully-connected*, are doing better than the other ones, and if some specific parametrization could improve less performing ones' results. Also, the GSP-base model's complexity could be investigated to make it possible to run on real-time devices for anomalous sound detection[1]. .

REFERENCES

[1] Abolfazl R., Xiwen Ch., Huayu L., Hao W., Brendan R., Yan Ch., and Hongbin Y., "Deep learning serves traffic safety analysis: A forward-looking review," *IET Intelligent Transport Systems*, vol. 17, no. 1, pp. 22–71, 2023.

[2] Z. Mnasri, S. Rovetta, and F. Masulli, "Anomalous sound event detection: A survey of machine learning based methods and applications," *Multimedia Tools and Applications*, pp. 1–50, 2021.

[3] F. Aurino, M. Folla, F. Gargiulo, V. Moscato, A. Picariello, and C. Sansone, "One-class svm based approach for detecting anomalous audio events," in *ICINCS*. IEEE, 2014, pp. 145–151.

[4] P. Foggia, N. Petkov, A. Saggese, N. Strisciuglio, and M. Vento, "Audio surveillance of roads: A system for detecting anomalous sounds," *IEEE trans. on int. trans. sys.*, vol. 17, no. 1, pp. 279–288, 2015.

[5] Z. Mnasri, S. Rovetta, and F. Masulli, "Anomalous sound event detection based on one-class classification using variational autoencoders and interval type-2 fuzzy sets," in *EUSIPCO*. IEEE, 2023, pp. 171–175.

[6] A. Ortega, P. Frossard, J. Kovačević, J. Moura, and P. Vandergheynst, "Graph signal processing: Overview, challenges, and applications," *Proceedings of the IEEE*, vol. 106, no. 5, pp. 808–828, 2018.

[7] L. Akoglu, H. Tong, and D. Koutra, "Graph based anomaly detection and description: a survey," *Data mining and knowledge discovery*, vol. 29, pp. 626–688, 2015.

[8] J. H. Giraldo, S. Javed, and T. Bouwmans, "Graph moving object segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 5, pp. 2485–2503, 2020.

[9] A. Sodemann, M. Ross, and B. Borghetti, "A review of anomaly detection in automated surveillance," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 6, pp. 1257–1272, 2012.

[10] J. H. Giraldo and T. Bouwmans, "GraphBGS: Background subtraction via recovery of graph signals," in *ICPR*. IEEE, 2021, pp. 6881–6888.

[11] H. Phan, M. Krawczyk-Becker, T. Gerkmann, and A. Mertins, "Weighted and multi-task loss for rare audio event detection," in *ICASSP*. IEEE, 2018, pp. 336–340.

[12] C. Kao, M. Sun, W. Wang, and C. Wang, "A comparison of pooling methods on lstm models for rare acoustic event classification," in *ICASSP 2020*. IEEE, 2020, pp. 316–320.

[13] S. Rovetta, Z. Mnasri, and F. Masulli, "Detection of hazardous road events from audio streams: An ensemble outlier detection approach," in *EAIS*. IEEE, 2020, pp. 1–6.

[14] B. Schölkopf, R. C. Williamson, A. Smola, J. Shawe-Taylor, and J. Platt, "Support vector method for novelty detection," *Advances in neural information processing systems*, vol. 12, pp. 582–588, 1999.

[15] Y. Kawaguchi, "Anomaly detection based on feature reconstruction from subsampled audio signals," in *EUSIPCO*. IEEE, 2018, pp. 2524–2528.

[16] Q. Wei and Y. Liu, "Auto-encoder and metric-learning for anomalous sound detection task," Tech. Rep., DCASE2020 Challenge, Tech. Rep, 2020.

[17] Nathanaël Perraudin, Johan Paratte, David Shuman, Lionel Martin, Vassilis Kalofolias, Pierre Vandergheynst, and David K Hammond, "Gspbox: A toolbox for signal processing on graphs," *arXiv e-prints*, pp. arXiv–1408, 2014.

[1]The code is available at https://github.com/zied-mnasri/ASD-GSP