Pattern Anal Applic (2012) 15:19–27
DOI 10.1007/s10044-011-0250-x

THEORETICAL ADVANCES

Dimensionality reduction for detection of moving vehicles

A. Averbuch · N. Rabin · A. Schclar · V. Zheludev

Received: 3 August 2009 / Accepted: 28 September 2011 / Published online: 13 October 2011
© Springer-Verlag London Limited 2011

Abstract Automatic acoustic-based vehicle detection is a common task in security and surveillance systems. Usually, a recording device is placed in a designated area and a hardware/software system processes the sounds that are intercepted by this recording device to identify vehicles only as they pass by. An algorithm, which is suitable for online automatic detection of vehicles, which is based on their online acoustic recordings, is proposed. The scheme uses dimensionality reduction methodologies such as random projections instead of using traditional signal processing methods to extract features. It uncovers characteristic features of the recorded sounds without any assumptions about the structure of the signal. The set of features is classified by the application of PCA. The microphone is opened all the time and the algorithm filtered out many background noises such as wind, steps, speech, airplanes, etc. The introduced algorithm is generic and can be applied to various signal types for solving different detection and classification problems.

Keywords Dimensionality reduction · Detection of moving vehicles · PCA · Diffusion maps

A. Averbuch (✉) · V. Zheludev
School of Computer Science, Tel Aviv University,
69978 Tel Aviv, Israel
e-mail: amir@math.tau.ac.il

N. Rabin
Department of Mathematics, Program in Applied Mathematics,
Yale University, New Haven, CT 06510, USA

A. Schclar
School of Computer Science, The Academic College
of Tel-Aviv-Yafo, 61083 Tel Aviv, Israel

1 Introduction

Moving vehicles produce typical sounds that are mainly influenced by their engine vibrations and the friction between the tires and the road. Airplanes, helicopters, wind, steps and speech create sounds that have different acoustic features when compared to vehicles. Similar vehicles types produce similar sounds, however, it is not a trivial task to identify similar vehicles that travel in diverse speeds, in various distances from the recording device and on different types of roads (land, asphalt, etc.). Our goal is to separate vehicles and non-vehicles sounds by analyzing their dynamic acoustic sounds. The recording device is ON all the time.

Every sound emitting device can be characterized according to acoustic features of the sounds it produces. These characteristic features are referred to as *acoustic signatures* and are used to differentiate vehicles and non-vehicles sounds. Usually, these signatures are analyzed by traditional signal processing methodologies. The proposed scheme uses ideas that come from compressed sensing [1, 2, 3] to uncover dominating features of an unknown acoustic signal. The short-term dynamics of the acoustic signal is treated as a point $x \in \mathbb{R}^m$. It is correlated with approximately $\log N$ random vectors in $\mathbb{R}^m$, where N is the total number of points. The outcome of this process is a set of features that is further processed to obtain an acoustic signature that differentiates it from others.

The algorithm has two phases: offline training and online detection. In the training phase, the data, which consist of vehicle and non-vehicle recordings, are analyzed and features that characterize it are extracted to produce their acoustic signatures. These signatures are used during the online detection phase. The proposed algorithm is generic and can be tailored to different tasks that need to separate between different clusters regimes.

The rest of this paper is organized as follows: Sect. 2 reviews previous work related to the problem at hand. Sect. 3 gives a short overview of the dataset and the algorithm goals. Sect. 4 outlines the structure of the proposed algorithm and in sect. 5 we described it in details. Experimental results are given in sect. 6.

2 Related work

Several papers deal with the problem of separating between vehicle and non-vehicle sounds. Most of them describe systems for a military context.

Extraction of acoustic features using the discrete wavelet transform is described in [4]. The feature vectors were compared to reference vectors in a database using statistical pattern matching to determine vehicle type from which the signal originated. The discrete cosine transform was applied in [5] to signals and a time-varying autoregressive modeling approach was used for their analysis. A system, which is based on wavelet packets coefficients in order to discriminate between different vehicles types, is described in [6]. Classification and Regression Trees (CARTs) were used for the classification of new unknown signals. In a later paper [7], the same authors used similar methods with a multiscale local cosine transform applied to the frequency domain of the acoustic signal. The classifier was based on the "Parallel Coordinates" [8, 9] methodology. Another recent paper [10] distinguishes between vehicles and background using wavelet packet coefficients with a procedure of random search for a near-optimal footprint. In [11], wavelet packet coefficients follwed by the application of Diffusion Maps [12], was used for vehicle classification. The"eigenfaces method" [13], which was originally used for human face recognition, to distinguish between different vehicle sound signatures, was used in [14]. The data were decomposed into a series of short-time frames. Then, each frame is transformed into the frequency domain. Classification is done by projecting the new frames on the principal components that were calculated for a known training set. Comparison between several speech recognition techniques for classification of vehicle types was presented in [15]. These methods were applied to short-time Fourier transform of the vehicles' acoustic signatures. Different types of moving vehicles in a wireless environment, which includes acoustic and seismic sensors, were classified in [16]. Each sensor extracted features by the application of the FFT. The averaged low frequencies values are saved. A local classifier like K-nearest neighbors, maximum likelihood or SVM classified the signal at each sensor. Then, a global fusion process classifies the final signal. A remote netted acoustic detection system for detection and classification of targets at tactical ranges was described in [17]. Hamming window and FFT were applied to windows at each sensor. Uniformly spaced beams were formed and frequency peaks were marked. The signal was classified according to harmonic lines that were generated from the frequency peaks. Multiple hypothesis tracking and Kalman filter algorithms were used for real time target tracking.

Because of the military context of these applications, there is not a benchmark dataset that is commonly used. The comparison between the work that has been done is difficult. The datasets that were used in the different papers were taken at different sample rates and the experimental conditions were not alike. In several papers the settings are different, the vehicles are classified using an array of sensors rather than a single one, this makes the classification task easier.

3 Structure of the dataset and problem description

The dataset contained almost 100 recordings that were collected in several different settings. The recordings sample rate was 48,000 samples per second, they were downsampled to 2,000 samples per seconds. In some of these settings the recoding device was close to the road (5–10 m), while in other settings it was placed further away from the road (up to 100 m). Most of the recordings were done when the vehicles traveled on an asphalt road, in one setting the vehicles traveled on a sand road. In addition, in some of the settings the road has sparse traffic, while other settings recorded busy roads with vehicles traveling at higher speeds. These varied settings made the classification task harder, and the designed algorithm, although is it generic, was constructed in a way that best utilized varied dataset and achieve good results even when applied to data in a low sample rate like 2,000 Hz. With this wide-ranging structure of the dataset, methods like eigenfaces, which was introduced in [14] were less promising. In [14], the training set was constructed under the assumption that there are a lot of samples of the same class, i.e., from the same kind of vehicle, recorded under similar conditions.

Furthermore, the algorithm presented here was to be constructed so that it will be suitable running on a small portable device, which would do the online processing and could be left unsupervised in a target location. This constraint led to utilizing the newly introduced random projection method, as an efficient tool for feature extraction, which does not require heavy processing. In addition, this paper gives some comparison between the use of random projections as opposed to more traditional signal processing tools like wavelets, and more sophisticated dimensionality reduction methods like Diffusion Maps, which were used in [10, 11].

4 Structure of the algorithm

An algorithm, which classifies acoustic signals and filters out non-related sounds, is proposed. The algorithm consists of two consecutive phases:

1. A learning phase in which acoustic signatures (features) are extracted from sample recordings whose classification is known.
2. A classification phase, which processes every newly arrived acoustic signal in order to determine according to the previously constructed acoustic signatures whether or not it is a vehicle.

The learning phase analyzes a known sample set of recordings $\mathrm{TSS} = \{s_i\}_{i=1}^{\tau}$ whose classifications are known a-priori where s_i is a recording of length $|s_i|$ and τ is the number of signals in the training set. The signals do not necessarily have the same size. Each signal s_i is decomposed into overlapping segments $W_i = \left\{w_j^i\right\}$ that are referred to as *windows*. A window size is $l = 2^r, r, l \in \mathbb{N}$. The windows are grouped into a single set $\Omega = \bigcup_{i=1}^{\tau} W_i$. For notational convenience, a single index is used in w_j^i and the output is denoted by $\Omega = \left\{w_j\right\}_{j=1}^{n_w}$ where the total number of windows resulting from the decomposition of all the signals is $n_w \triangleq |\Omega|$.

Following the decomposition, features are extracted from every window by the application of random projections. The classification phase does not process the entire signal in a batch mode but a short segment at a time. This fact along with the high efficiency of the algorithm render this classification to fit real-time applications.

5 The Classification algorithm

The applicability of dimensionality reduction via random projections was proved in [18]. Specifically, it was shown that N points in N dimensional space can almost always be projected into a space of dimension $C\log N$ where the ratio between distances and error (distortion) is controlled. Bourgain [19] showed later that any metric space with N points can be embedded by a bi-Lipschitz map into an Euclidean space of $\log N$ dimension with a bi-Lipschitz constant of $\log N$. Various randomized versions of this theorem are used for protein mapping [20] and for the reconstruction of frequency-sparse signals [2, 1, 3]. Recently, machine learning techniques used in compressed sensing (random projections) were used for finding the intrinsic dimension of the data. It can replace traditional feature extraction methods to go from high dimensional to low dimensional space. In this case, the projected data become the feature space and these features are classified by some classification algorithm. Manifolds construction for learning, which is based on random projections, is given in [21]. Random projections were used in [22] to extract features from face images. In addition, random projection can be added as a dimensionality reduction step to algorithms that select features in different ways. For example, noisy speech signals in [22] were classified and random projections were used as a tool to reduce the data dimension to get faster computational results.

We assume that the acoustic data signals have some sparse representation. The goal is to find the most important coefficients, which contain information that will discriminate between input classes. The application of random projections to the signal is used to extract the dominating features, which will later separate between vehicles and non-vehicles. The PCA later saves the features that are most important for the separation process. The use of random projections in this algorithm is equivalent to applying wavelets (or any other method for acoustic feature extraction) followed by the application of PCA.

In order to reduce the dimensionality by using random projections of a dataset $\Gamma = \{x_1, x_2, \ldots, x_n\}$, which holds column vectors in $\mathbb{R}^m$, a random matrix $\Upsilon = (\rho_{ij}), i = 1, \ldots, q, j = 1, \ldots, m$, is generated, where q is the dimension of the target reduced space. Two common choices for generating a random matrix are:

1. The columns of Υ are uniformly distributed on the q dimensional unit sphere.
2. The column elements of Υ are chosen from a Bernoulli $+1/-1$ distribution and the columns are l_2 normalized to have length 1.

The embedding $\widetilde{x}_i$ of x_i into a low dimensional space is obtained by

$$\widetilde{x}_i \triangleq \left(\Upsilon \cdot x_i^T\right)^T, i = 1, \ldots, n \tag{5.1}$$

where T denotes the transpose of a vector/matrix and $\cdot$ is an inner product.

Following the random projection stage, the classifier concatenates every μ consecutive projected windows and further reduces the dimensionality by applying Principal Component Analysis (PCA) to this concatenation. PCA, which is common way for dimensionality reduction of high dimensional data, projects the data onto the direction where the variance of the data is maximal. The classification is done in the dimension-reduced space. Thus, two dimensionality reduction steps are applied to the data. This, assures a better compaction of the data than if a single dimensionality reduction technique had been used.

5.1 The Learning phase

The learning phase uses the random projection methodology in order to extract features from every input window $w_j \in \Omega$. Algorithm 1 outlines the main steps in the *learning* phase.

Algorithm 1 Learning Phase

1. Every signal is decomposed into overlapping windows.
2. The windows are transformed into the frequency domain via the application of the Fourier transform.
3. Dimensionality reduction via random projections: each window is projected onto a given number n_{RM} of random generated bases.
4. Paths from the random projections are constructed. A path contains the random projections of μ consecutive windows where μ is a given parameter.
5. Dimensionality reduction of the paths via the application of PCA.

In the following, we describe steps 2–5 in details.

Step 2: Since the data are changing with time, the comparison of windows in the original time domain is difficult. Transforming the signals from the time domain into ones defined in the frequency domain, is a common step in signal processing applications, where the input is non-stationary, see [14], [16] and [15]. For this type of acoustic data, the acoustic signature of a signal is better seen in its frequency domain. The fast Fourier transform (FFT) is applied to each window w_j from step 1. The magnitudes of the frequencies are saved. Furthermore, the dynamic range is reduced by taking the logarithm of the magnitudes (a small constant is added to avoid taking a logarithm of zero-magnitude frequencies). The output of this step is denoted by $U \triangleq \{u_j\}_{i=1}^{n_w}$, $u_j \in \mathbb{R}^h, h = \frac{l}{2}$, where l is the window size.

Step 3: A number of random matrices $\mathrm{RM} = \left\{\Upsilon^i\right\}_{i=1}^{n_{\mathrm{RM}}}$ are generated where Υ^i is the i^{th} matrix of size $r \times h$. The dimension of the set U is reduced by projecting it using every matrix in RM, as described in Eq. 5.1. Every projection, which uses Υ^i, produces a single embedding into a dimension-reduced space. The *random projection* of U onto a random basis Υ^i is denoted by $\widetilde{U}^i \triangleq \left\{\widetilde{u}_j^i\right\}_{j=1}^{n_w}$ where $\widetilde{u}_j^i \in \mathbb{R}^r$. Each projection $\widetilde{u}_j^i$ describes the acoustic signature of w_i. A single projection is referred to as a dimension-reduced-window (DRW) and the set of all random projections on RM is denoted by $\widetilde{U} = \left\{\widetilde{U}^i\right\}_{i=1}^{n_{\mathrm{RM}}}$.

Step 4: Given a random projection $\widetilde{U}^i = \left\{\widetilde{u}_j^i\right\}_{j=1}^{n_w}$, all sequences of μ consecutive DRWs are constructed. These sequences are referred to as *paths*. A path captures the short-term dynamics of a signal at a specific time. Furthermore, a path is more robust to local noise (such as a wind gust) than a single window since the duration of the dynamics it captures is longer than that of a single window. This construction is done separately for each subset of DRWs according to the original signal classification. Specifically, every vector $\widetilde{u}_j^i$ is labeled according to the class of its corresponding signal s_k. As mentioned above, the classifications of the signals, which are analyzed during the learning phase, are known a-priori, so a label is associated with each signal. $\widetilde{U^i}$ is separated according to the labels of the DRWs and the paths are constructed in each set by concatenating μ sequential DRWs. Denote the paths constructed from the DRW of all the signals, which were obtained by the random matrix Υ^i, by $P^i \triangleq \{\widetilde{p}_j^i\}_{j=1}^{n_w-\mu+1}$, where $\widetilde{p}_j \in \mathbb{R}^{r \cdot \mu}$. The output of this step is the set $P = \{P^i\}_{i=1}^{n_{\mathrm{RM}}}$ that consists of n_{RM} learning sets that contain the short-term dynamic paths of the acoustic signatures. These sets are organized according to the classification (labels) of the paths.

Step 5: Let P^i be the paths constructed from the DRW of all the signals via the random matrix Υ^i. The paths in P^i are shifted to be centered around the origin. PCA is applied to the set P^i. The projection of the dataset P^i onto the first k principle components yields $Q^i \triangleq \{\widetilde{q}_j^i\}_{j=1}^{n_w-\mu+1}$, where $\widetilde{q}_j \in \mathbb{R}^k$. This step is performed for every set of paths P^i, $i = 1, \ldots, n_{\mathrm{RM}}$, that was produced in step 3. Thus, n_{RM} low-dimensional learning-sets, $Q = \{Q^i\}_{i=1}^{n_{\mathrm{RM}}}$, are created by projecting the paths, which were created in step 3, onto the PCA bases that were constructed in this step.

The flow of the learning algorithm is presented in Fig. 1.

5.2 The Classification phase

The classification phase is performed online. There is no need to wait for the entire signal to be received. In order to classify the signal *at time t*, the algorithm only needs the path that ends at time t, i.e., the μ consecutive overlapping windows of size l that immediately preceded time t. The

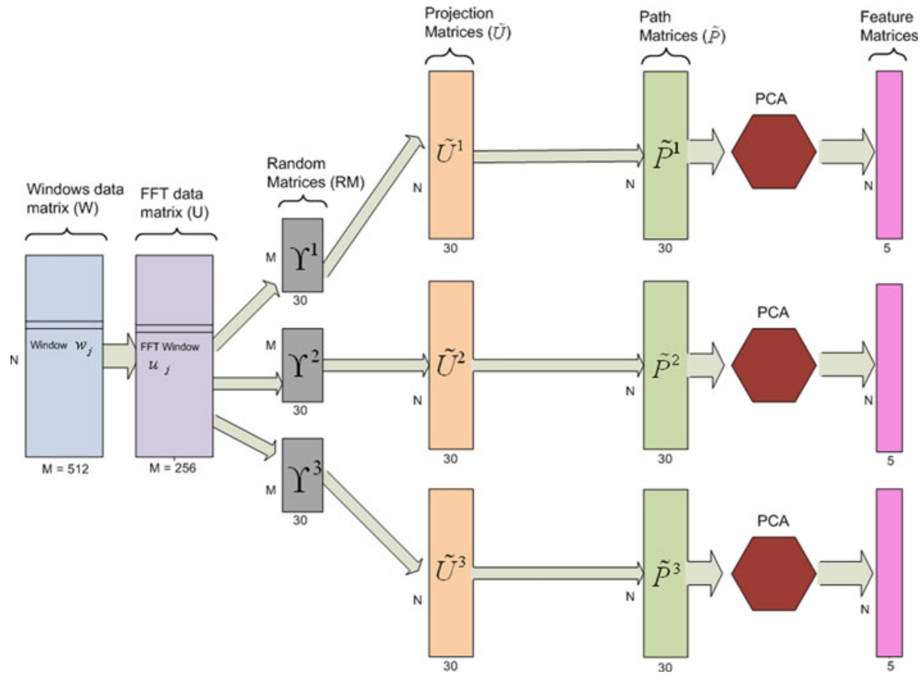


Fig. 1 Flow of the learning algorithm

values of μ and l are the same as those used in the learning phase. Let $\sigma_t = (\sigma(t-v+1), \sigma(t-v+2), \ldots, \sigma(t))$ be a sequence of v signal values that were received up to time t where $\sigma(x)$ is the signal's value that was captured at time x. σ_t is decomposed into μ overlapping windows $\{\omega_j\}_{j=1}^{\mu}$ of size l. In order to classify $\{\omega_j\}_{j=1}^{\mu}$, an algorithm, which is similar to Algorithm 1 in sect. 5.1, is employed. Algorithm 2 outlines the steps for classifying σ_t.

Here is a detailed description of each step in Algorithm 2.

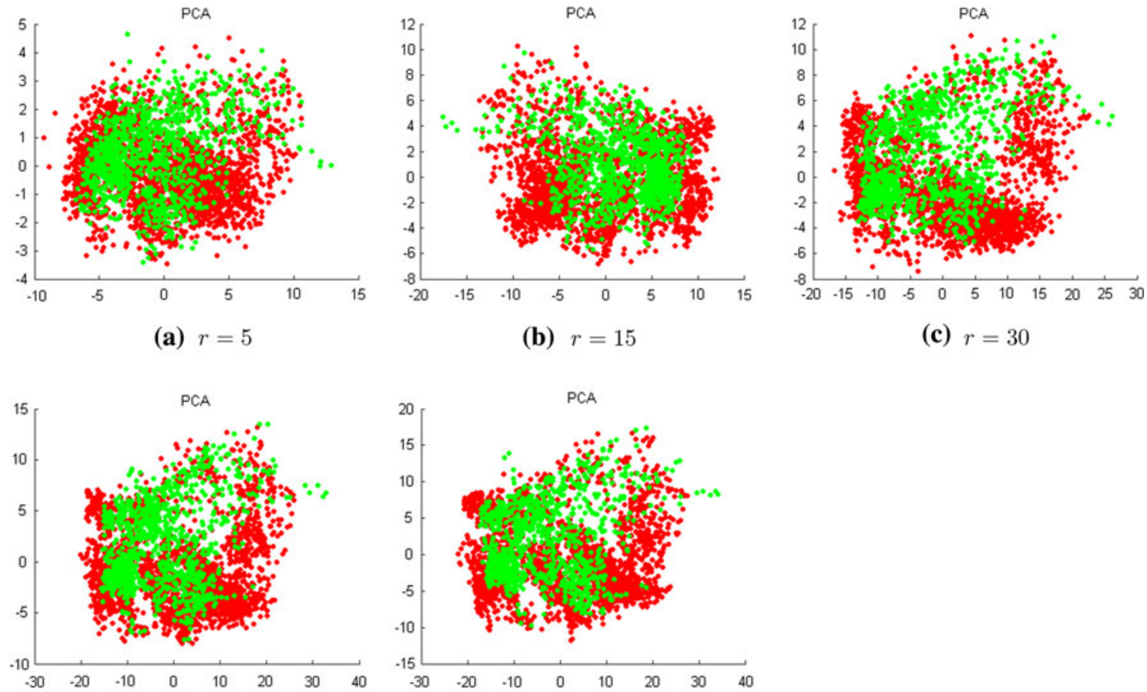
Step 1: The FFT is applied to each window in $\{\omega_j\}_{j=1}^{\mu}$. As in step 2 of Algorithm 1, the logarithm magnitudes of the frequencies are saved and the result is denoted by $\{\upsilon_j\}_{j=1}^{\mu}$.

Step 2: The dimensionality of $\{\upsilon_j\}_{j=1}^{\mu}$ is reduced by randomly projecting it using *all* the random matrices $\mathrm{RM} = \{\Upsilon^i\}_{i=1}^{n_{\mathrm{RM}}}$ that were generated in step 3 of Algorithm 1. The projection via a single matrix Υ^i produces a set of reduced dimension vectors $\left\{\tilde{\upsilon}_j^i\right\}_{j=1}^{\mu}$.

Algorithm 2 Classification phase

---

1. Application of FFT.
2. Application of dimensionality reduction using random projections via the matrices that were generated in step 3 in Algorithm 1.
3. Construction of a path from the output of step 2.
4. Application of dimensionality reduction using the principle components that were calculated in step 5 of Algorithm 1.
5. Classification of the newly arrived sample according to its nearest neighbor in the reduced space produced by the PCA.



(a) $r = 5$ **(b)** $r = 15$ **(c)** $r = 30$

(d) $r = 45$ **(e)** $r = 60$

Fig. 2 The embedding of the training set via the first two PCA coordinates, after random projecting the dataset to different number of random matrices, r. The dark points close to the boundaries embed windows that belong to vehicle recordings and the bright points embed windows that belong to background recordings. It can be seen that the separation improved as r is raised from 5 to 30, and stays quite stable for $r = 30$, 45 and 60

Step 3: For each single matrix Υ^i, the vectors $\left\{\widetilde{\upsilon}_j^i\right\}_{j=1}^{\mu}$ are concatenated into a path ϕ^i. Thus, the output of this step is a set of n_{RM} paths $\{\phi^i\}_{i=1}^{n_{\mathrm{RM}}}$.
Step 4: The set of paths $\{\phi^i\}_{i=1}^{n_{\mathrm{RM}}}$ is projected on the first k principal components that were calculated in step 5 of Algorithm 1. These embeddings are denoted by $\{\varphi^i\}_{i=1}^{n_{\mathrm{RM}}}$.
Step 5: Let Q^i be a low dimensional learning set that was generated in Algorithm 1 using the random matrix Υ^i and let φ^i be the new embedded signal that was produced using the same random matrix Υ^i. The δ nearest neighbors of φ^i from the set Q^i are found and their labels are saved. The classification of the new arrived signal is determined according to the label with the highest number of occurrences within the group of the nearest neighbors that were gathered from the entire learning set $Q = \{Q^i\}_{i=1}^{n_B}$.

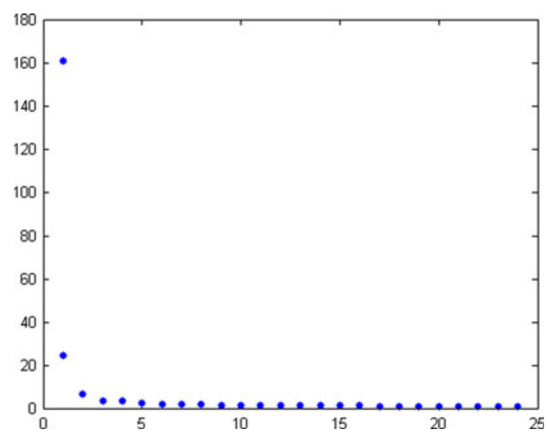


Fig. 3 The distribution of the PCA eigenvalues

6 Experimental results

The algorithm, which is described in Sects. 4 and 5, is applied to a sample set of 180 short recording. Each recording, which belongs to the sample set, was identified by an expert as either a vehicle recording or a recording that contains background noises such as helicopters, speech wind and airplanes to name some. In the given sample set, 120 recordings were vehicles (cars, trucks and vans), while the other 60 recordings were of background. The recordings were sampled at 2,000 Hz.

In order to analyze the performance of the algorithm, we applied a fourfold validation. In each iteration, 90 vehicle recordings and 45 non-vehicle recordings were used as a

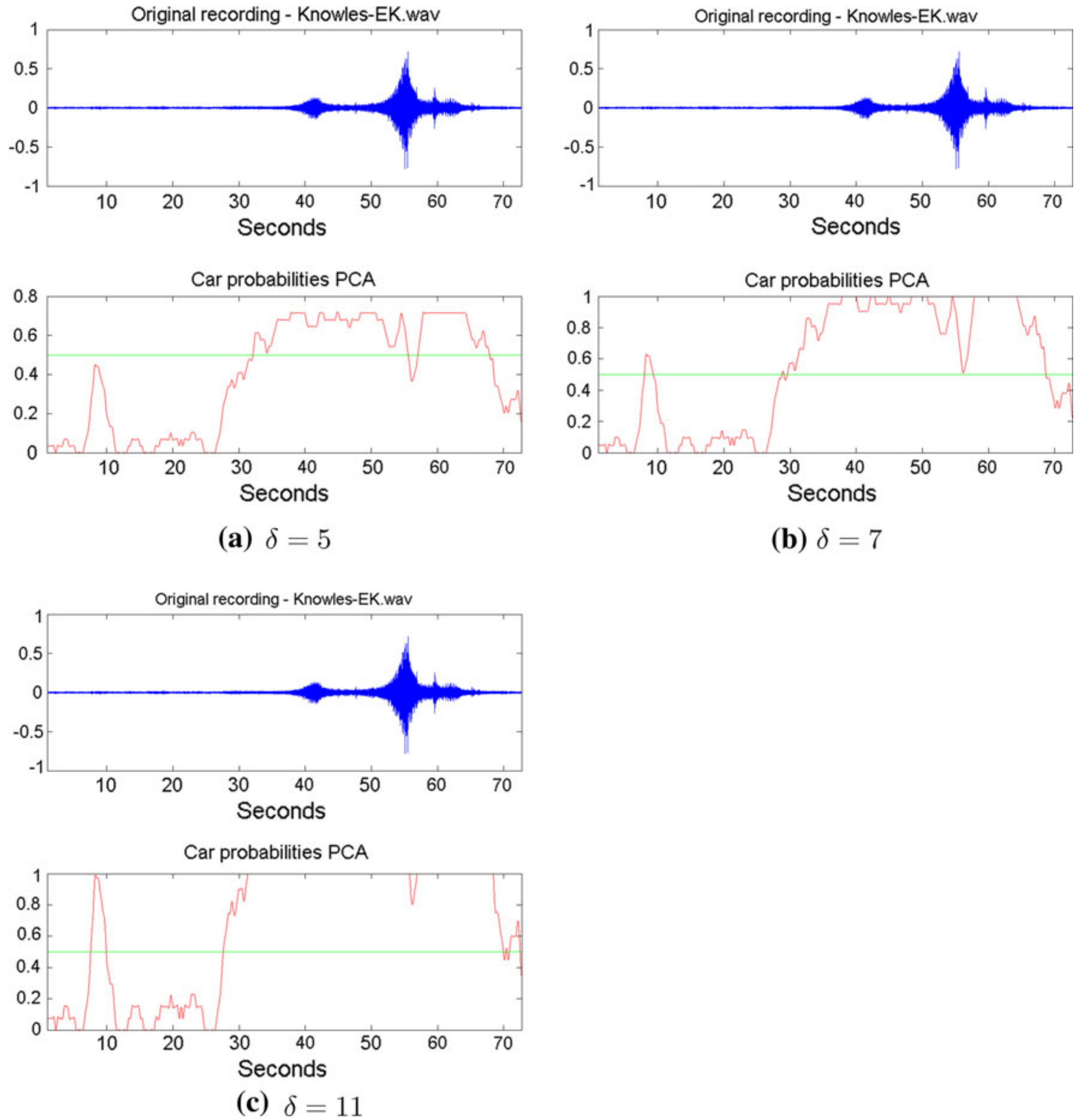


Fig. 4 Classification of a car and a truck that pass by at $t = 40$ and 50 s with three different values of δ. It can be seen that when δ is set to be 5 in **a**, the vehicle classification percentage is smaller than 1 at all times. On the other hand, setting δ to be 11 in **c** generates a false-positive classification at the beginning of the recording. Setting δ to 7 in **b** yields the best results for this recording an for a larger set of simple test recordings

training set. The test set at each iteration consisted of 30 vehicle and 15 non vehicle recordings.

The following parameters were used for the learning and classification phases: Window size, l, was set to 1,024 and the overlap between consecutive windows was 50%. The number of random matrices, which were generated in Algorithm 1 step 3, was RM $= 3$. The number of random vectors in each random matrix was $r = 30$. Using Johnson Lindenstrauss lemma [18] for random projections, the dimension is reduced to $O(\frac{\ln(512)}{\epsilon^2})$, $0 < \epsilon < 1$. Setting ϵ to 0.5 implies that the dimension can be reduced to approximately 30. Figure 2 shows how the embedding of the training set changes according to different values of r. Raising r from 5 to 15 and then to 30 improves the separation. Raising it beyond 30, to 45 and 60 does not improve much. This implies about the dimension of the feature space.

The path length μ, defined in algorithm 4 step 4, was set to 5. These parameters depend on the sample rate and window length and it should grow if l is set to 512. The number of principal components used in step 5 in Algorithm 1 was $k = 5$. Figure 3 shows the spectrum of the PCA eigenvalues. It can be seen that the first two PCA coordinates are the most important, but setting $k = 5$ improved the results. These parameters were determined empirically after testing various values of k on a set of vehicles recordings that were taken in convenient environmental conditions.

A new point, which is embedded online by Algorithm 2 into principal components, is classified according to its $\delta = 7$ nearest neighbors. These parameters were determined empirically. It was tested on a set of vehicles that were easy to detect. Figure 4 shows the classification of a car and a truck that pass by at $t = 40$ and $t = 50$ seconds with three different values of δ.

Table 1 presents the averaged results in percentages of four confusion matrices. We see that the correct vehicles detection rate is very high while the correct background detection rate is lower. The cause for this is both the large recoding types diversity that are classified as background and the relatively small number of background recordings. The correct average detection rate for the test datasets is 86 %.

The results are compared with another vehicle detection algorithm, which uses wavelets rather than random projections as a feature extraction method. The classification is achieved via the application of PCA. The feature extraction steps are similar to those that were used in [11]. The classification in [11] was achieved by the application of Diffusion Maps. In order to compare directly with the presented method, we use PCA as a classifier in both algorithms.

1. Application of the wavelet packet transform, which uses spline wavelet of order 4, to each acoustic window.
2. Calculation of the energy distribution of the wavelet coefficients by summing the coefficients in every frequency band in each block.
3. Every 5 consecutive segments are averaged to achieve noise reduction.

The fourfold cross validation is applied to the same sample set. The confusion matrix is presented in Table 2. The average correct detection rate for the test data sets is 84%.

Table 1 An average of the four confusion matrices, which are generated for each fourfold application of the algorithm

		Predicted class	
		vehicle (%)	back (%)
True class	vehicle	95	5
	back	23	77

The values in the confusion matrix are given in percentages

Table 2 An average of the four confusion matrices, which are generated for each fourfold application of the wavelet-based algorithm

		Predicted class	
		vehicle (%)	back (%)
True class	vehicle	88	12
	back	20	80

The values in the confusion matrix are given in percentages

These results emphasize the strength of the random projections as method for feature extraction. The results using random projections are slightly better then those of the algorithm which uses wavelets.

7 Conclusions and future work

We presented a two-phase algorithm that detects vehicles according to their acoustic characteristics. Every acoustic signal was decomposed into overlapping windows and dominating features were extracted from each window using random projections. Short-term dynamic paths were then constructed from sequences of features that were extracted from consecutive windows. In order to detect the vehicles, these paths were embedded into a lower dimensional space using PCA. The online classification of new signals was obtained by employing similar steps.

The results, which were presented in sect. 6, were based on a relatively small training set. The experiments indicate that the accuracy of the classification is affected by a number of factors:

The size of the training set: using a larger number of recordings during the learning phase provides a more reliable training set which results in a more accurate detection.

Coverage of the test sample set: coverage of the test sample set: The training set should include a large variety of background noises that are typical to the detection area or otherwise discrimination between background noises, which are not included in the training set, is not guaranteed. This fact is reflected in the results that are presented in Tables 1 and 2. The correct vehicles detection rate is high, since the dataset included a large number of vehicle recordings examples. The large variety of background noises, which may change with time and are more difficult to gather for the training set, affected the correct detection rate of non-vehicle acoustic recordings.

The introduced algorithm is generic and can be applied to various signal types for solving different classification problems.

Acknowledgments This was supported by the Israeli Ministry of Science and Technology.

References

1. Tsaig Y, Donoho DL, Drori I, Strack JL (2006) Sparse solution of underdetermined linear equations by stagewise orthogonol machine pursuit. Technical report No. 2006-2 Departement of Statistics, Stanford University, April 2006
2. Donoho DL (2006) Compressed sensing. IEEE Trans Inf Theory 52(4):1289–1306
3. Romberg J, Candes E, Tao T (2006) Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. IEEE Trans Inf Theory 52(2)
4. Meitzler T, Gerhart GR, Choe HC, Karlsen RE, Gorsich D (1996) Wavelet-based ground vehicle recognition using acoustic signals. Proc of the SPIE 2762:434–445
5. Eom KB (1999) Analysis of acoustic signatures from moving vehicles using time-varying autoregressive models. Multidimens Syst Signal Process 10:357–378
6. Zheludev V, Averbuch A, Hulata E (2004) Identification of acoustic signatures for vehicles via reduction of dimensionality. Int J Wavelets Multiresolut Inf Process 2(1):1–22
7. Zheludev V, Averbuch A, Hulata E, Kozlov I (2001) A wavelet packet algorithm for classification and detection of moving vehicles. Multidimens Syst Signal Process 12(1):9–31
8. Inselberg A (1998) Visual data minint with parallel coordinates. Comput Stat 13(1):47–63
9. Inselberg A, Dimsdale B (1994) Multi-dimensional lines i: Representations, ii: proximity and applications. SIAM J on Appl Math 54(2):559–577
10. Averbuch A, Zheludev V, Rabin N, Schclar A (2009) Wavelet-based acoustic detection of moving vehicles. Multidimens Syst Signal Process 20(1):55–80
11. Schclar A, Averbuch A, Rabin N, Zheludev V, Hochman K (2010) A diffusion framework for detection of moving vehicles. Digit Signal Process 20:1
12. Lafon S, Coifman RR (2006) Diffusion maps. Appl Comput Harmon Anal Special issue Diffus Maps Wavelets 21:5–30
13. Sirovich L, Kirby M (1987) Low-dimensional procedure for the characterzation of human faces. J Opt Soc Aner Soc A 4(3): 519–524
14. Siegel M, Wu H, Khosla P (2004) Vehicle sound signature recognition by frequency vector principal component analysis. IEEE Trans instrum meas 48(5)
15. Munich ME (2004) Bayesian subspace method for acoustic signature recognition of vehicles. In: Proc. of the 12th European signal processing conf. EUSIPCO
16. Marco F. Duarte, Yu Hen Hu (2004) Vehicle classification in distributed sensor networks. J Parallel Distrib Comput 64(7): 826–838
17. Srour N, Robertson J (1995) Remote netted acoustic detection system: final report. Army Res Lab ART-TR-706
18. Johnson WB, Lindenstrauss J (1984) Extensions of lipshitz mapping into hilbert space. Contem Math 26:189–206
19. Bourgain J (1985) On lipschitz embedding of finite metric spaces in hilbert space. Israel J Math 52:46–52
20. Tishby N, Linial M, Linial N, Yona G (1997) Global self-organization of all known protein sequences reveals inherent biological signatures. J Mol Biol 268:539–556
21. Wakin M, Hegde C, Baraniuk R (2007) Random projections for manifold learning. Neural information processing systems (NIPS). Vancouver, Canada
22. Cranen B, Gemmeke JF (2008) Noise robust digit recognition using sparse representations. Interspeech 2008, Brisbane, September 2008