

Data Extraction with Signal Comparison for Forest Logging Prevention

Rohith Pandith M V

Department of Electronics & Communication
The National Institute of Engineering
Mysore, India

Prabhulingesh K

Department of Electronics & Communication
The National Institute of Engineering
Mysore, India

Shashank Hegde

Department of Electronics & Communication
The National Institute of Engineering
Mysore, India

Srishanth S Amin

Department of Electronics & Communication
The National Institute of Engineering
Mysore, India

Abstract— Due to an increase in deforestation nowadays, it has immensely impacted our ecosystem which is in turn causing global warming and major effects on biodiversity in our ecosystem.

The idea of our paper is to prevent tree logging by extracting sound of logging machine from sensors placed at specific ranges by performing cross correlation and other signal processing techniques between test signals obtained from the sensor and pre-fed signal at the nearest office of interest and hence avoiding tree logging activity by alerting respective authorities.

Keywords: cross correlation, power spectrum

I. INTRODUCTION

This paper proposes a concept of extraction of specific sound corresponding to the sound of different logging machines from sensors and performing various signal processing techniques along with extraction of various statistical parameters from the signals obtained from the sensors with the classification of them. The idea is to compare signals obtained from the sensor with pre-fed signals in the central office and if it is matched with any one of the signals in the central office it is then informed to respective authorities, as an illegal logging machinery might have entered the premises of the forest.

Our aim is to reduce the illegal tree logging in forests and national parks and assist the government to implement a simple, yet cost effective solution to nurture the biodiversity.

II. RELATED WORK

In the paper by József Kopják [1], The author has compared different data collecting methods like polling, synchronize broadcast response and merged data collection (MDC) and concluded which one is best among them. The IQRF technology is used in test sensor network, by using distance from network coordinator called virtual routing number (VRN). During polling, request packets are sent to each node, while optimization in second method is achieved by broadcasting packets to every node. MDC works by sending the measured sensor values in one response packet as response to a broadcast request. Fast response code (FRC) supports node which has no VRN number, with FRC being optimal among these methods.

In the paper by Gokay Saldamli [2], the node uses CO and NO gas sensors to detect fire. The wireless network sends an alert to the dashboard which displays sensor data along with its id which defines its precise location. Each node has Raspberry Pi to connect to the internet and LoRa technology used for communication between a node and central office. Dashboard displays readings from the sensor and an alert message is sent if the value crosses threshold value. A built-in library from java script allows automatic plot of these sensor values of a particular date which helps to better analyze sensor data at the end user.

In the paper by L.K. HEMA [3], identifying the logging of trees in forest by means of the vibration of illegal tree logging is discussed. The vibration sensor should have good selectivity because there is chance of mixing of environmental vibrations with intended vibration of signal. Each node contains a PIC microcontroller, vibration sensor and Zigbee transceiver. The accelerometer is used to measure vibration which changes its capacitance value whenever vibration occurs which is fed to PIC microcontroller and if vibration crosses threshold value it alerts the respective authorities. The disadvantage of this method is that vibration sensors must be placed on each tree to indicate a vibration during tree logging, which increases the system cost.

In the paper by Yu-Yan Chen [4], it aims to recognize the vibration of illegal logging events and the technique of illegal logging sound recognition. There are 3 type of nodes namely, gravity sensor (GSN), audio sensor nodes (ASN) and video sensor nodes (VSN) and also Fog-computing node with Lightweight SDN controllers (FLCs). The various sensor nodes measure the illegal activity if it crosses threshold and send the data to FLC. Atmega368P is used for controlling of nodes and Zigbee is used for data transmission. VSN is enabled only if either GSN or ASN is turned on and signal processing of this data is done only in time domain.

III. METHODOLOGY & IMPLEMENTATION

Each node consists of a STM32 controller, various sensors such as Temperature sensor (LM 35 sensor) and a gas sensor (MQ2 gas sensor) to detect forest fires, a power module with battery and solar panel, a sound sensor, microphone and a transmitter to send the data to the receiver at the central processor, as shown in Fig 3.1. The nodes are implemented in the forest in a hexagonal shape for maximum area coverage.

All tree logging machines have sound values higher than 60 dB, hence the microphone only turns on & captures sound higher than 60 dB, as sensed by the sound sensor and transmits the audio to the Transmitter via the microcontroller. Also, any unusual temperature or gas concentration values are sent to the transmitter for fire detection. The transmitter then sends the data received to the receiver at the Central Computer.

Each node will transmit data in a different channel using a unique frequency. The receiver will be scanning for incoming signals from the nodes at all the frequency ranges corresponding to the nodes. When a signal is received on a specific channel, the frequency can be mapped to the corresponding area from which the signal and possible danger of illegal machinery ingress came from. This is processed at the central processor

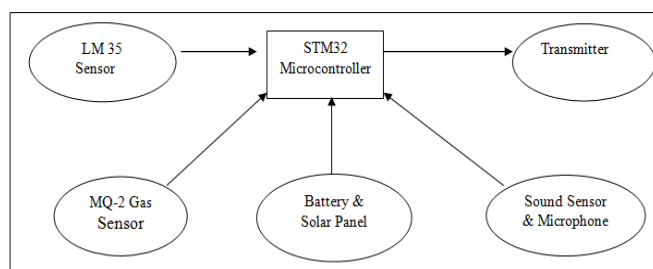


Fig 3.1 Connections at each node

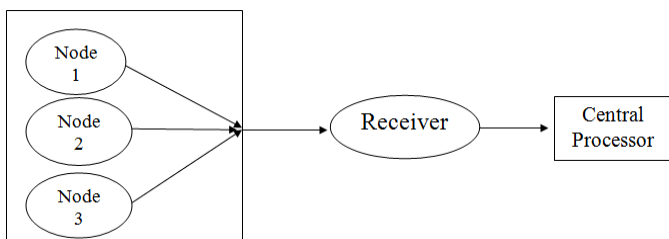


Fig 3.2 Node connection and transmission to central office

Then audio signal from the forest is compared with pre-fed signals to find the similarity between the audio signals. In order to find the similarity in time domain we have used cross correlation between the two signals. Power Spectrum can be used to find similarity in the frequency domain along with extraction of different statistical parameters from the signal which is compared with pre-fed signal. After extraction of parameters we will classify the signals with the help of k nearest neighbor algorithm. We have used GNU octave which is an open source software similar to MATLAB but it comes with different packages. We need to download packages depending upon our application.

As shown in Fig 3.3, we pre feed the system with various signals of a diverse set of logging machines. This is the training data. Signal features such as energy, skewness, ZCR etc. are extracted from the signals. We will thus obtain a database of statistical parameters of these logging machine signals.

Similarly, signal features of test signals obtained are extracted and compared with the signal statistics of the trained signals. Matching is done between the test signal and various trained signals with least Euclidean distance through

a K Nearest Neighbor (KNN) classifier, and an output is obtained which shows the probable logging machine of the test signal

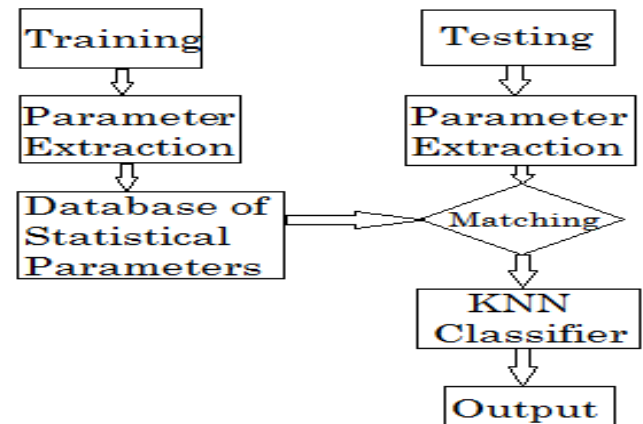


Fig 3.3 Block Diagram of system working in Octave

A. Correlation between Signals

Correlation between two signals means measuring the similarity between the signals and degree to which the two signals are correlated, and is defined by the term called correlation coefficient. There are various correlation coefficients which can be deployed for finding similarity between two signals. In this paper we have used Pearson Correlation coefficient which measures linear relationship between two signals.

If the signal is compared with delayed version of itself, then it is called auto correlation and resulting output will have conjugate symmetry and have a peak at zero lag or zero time instant, and if we normalize this output, we will get unitary power and null mean.

We have used xcorr function in GNU octave to find correlation between signals.

$[c, lag] = xcorr(x, y)$

c is the correlation matrix

lag is time difference between two signals

B. Pearson correlation coefficient

Pearson correlation coefficient measures linear relation between signals.

It is given by,

$$\text{correlation coefficient, } r = \frac{\text{cov}(x, y)}{\sigma_x * \sigma_y}$$

Where, numerator is covariance of two signal x and y denominator is product of standard deviation of two signal.

We know that standard deviation can't be negative so denominator is always positive and in the numerator the covariance is measure of how changes in one signal reflect in second signal. Thus, numerator can be positive or negative.

The value of correlation coefficient lies between 1 and -1, and value of correlation coefficient is 1 indicates signal is having perfect relationship between two signals and depending upon sign of its value it can be either positive or negative relationship.

C. Power Spectral Density

The Fourier transform of autocorrelation is defined as power spectral density (PSD). FFT provides us spectrum density

frequency of the time domain signal. The absolute value of FFT is squared is defined as PSD. The time series signal contains a power spectrum which extracts the frequency components of that power signal described. According to Fourier analysis, a number of discrete frequencies could be aggregated in any physical signal or in a spectrum of signals which are found over a continuous range. The statistical mean of a signal analyzed in its frequency domain, is called its spectrum. The PSD is the spectral energy distribution per unit time that would be found. The total power can be computed as summation or integration of spectral components. The uses of PSD tools include identification of oscillatory signals in time series data, along with their amplitude, and recognition at which frequency ranges the variations of signals are strong, which could be useful for analyzing the signal.

D. K Nearest Neighbor Algorithm

This algorithm is employed for the classification of the signal obtained from the forest with the pre-fed signal in the central system. This algorithm stores all available cases of different classes and classifies new unknown cases based on distance function namely Euclidean distance, which measures distance between unknown case and all other stored cases and classifies unknown case to the particular class depending upon least Euclidean distance among the stored cases. The classification is done by majority of least Euclidean distances which depends on number of stored cases, because we will select 'k' which is number of cases used for the classification by square root of number of stored cases.

IV. SIGNAL STATISTICAL PARAMETERS

A. Energy

The energy of signal is computed as magnitude square of each sample in the signal.

$$E_s = \sum_{n=-\infty}^{\infty} |x(n)|^2$$

B. Number of Zero Crossings

It is count at which sign changes in the signal. It is computed by the number of times the value of signal changes from the positive to negative or vice versa.

It can be interpreted as a presence of noise in a signal, so higher value of zero crossings means signal is more prone to noise.

Zero crossing is a fundamental property which is employed in classification of audio. It is extensively used in a various audio application in speech analysis and sound recognition.

C. Standard Deviation:

Standard deviation is a measure of dissimilarity for a set of values and lower the value indicates samples are close to arithmetic mean and higher the value indicates values are more diverse in the data set.

$$SD = \sqrt{\frac{1}{N-1} * \sum_{i=1}^N (x_i - \mu)^2}$$

Where, N= number of samples

x_i = current score or data set

μ = arithmetic mean

D. Coefficient of Variation:

The coefficient of variation helps in better understanding of values in the data set by relation between standard deviation and arithmetic mean. It is particularly used when two data set have distinct arithmetic mean.

$$\text{coefficient of variation} = \frac{\text{standard deviation}}{\text{arithmetic mean}}$$

E. Z-score (Standard score):

It tells the number of standard deviations by which the value of a current sample of dataset is above or below the mean value of data set or how far the current point from the mean of the whole data set. Thus, it can be used to compare different score that are taken from different test sample.

$$Z = \frac{x - \mu}{\sigma}$$

Where z is standard score

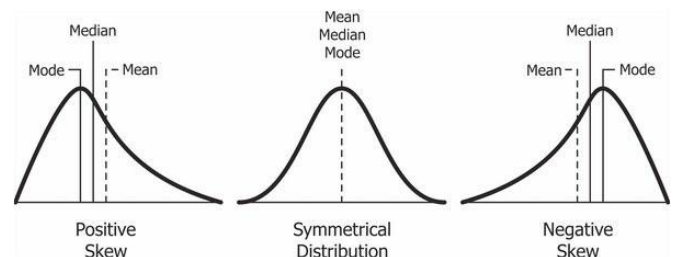
X is current sample of data set

μ is arithmetic mean of data set

σ is standard deviation of data set

F. Skewness:

It is measure of asymmetry of the probability distribution in a data set. The skewness can take positive, negative or zero value.



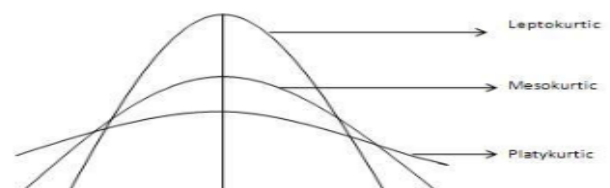
The curve is said to be highly skewed, if the value is less than -1 and greater than 1 and it can be called as moderately skewed if the value lies in between -1 and -0.5 or 0.5 and 1 and curve is said to be having symmetric distribution if value is between -0.5 and 0.5.

skewness: $s = v_3 / v_2^{3/2}$

where, $v_3 = \sum (x - \bar{x})^3 / n$ and $v_2 = \sum (x - \bar{x})^2 / n$

\bar{x} is the mean or average, n is the number of samples, v_3 is called the third moment of the samples. v_2 is the variance, the square of the standard deviation.

G. Kurtosis



It is measure of tail of distribution is compared with tail of normal distribution (which has kurtosis =3). The curve is said to be platykurtic, if kurtosis less than 3 and it is called as leptokurtic if kurtosis value is greater than 3.

Kurtosis is given by mathematically,

kurtosis: $k = v_4 / v_2^2$

where, $v_4 = \sum (x - \bar{x})^4 / n$ and $v_2 = \sum (x - \bar{x})^2 / n$
 \bar{x} is the mean or average, n is the number of samples, as usual. v_4 is called the fourth moment of the samples. v_2 is the variance, the square of the standard deviation

V. SIMULATIONS & RESULTS

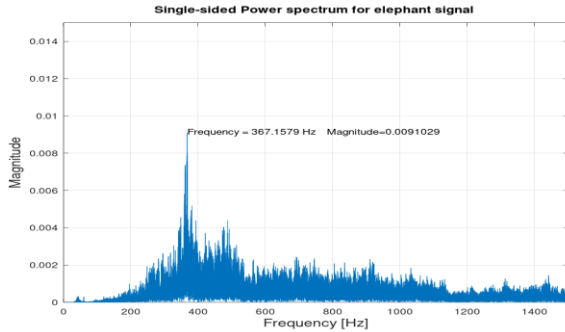


Fig 5.1 Single sided power spectrum of elephant trumpet

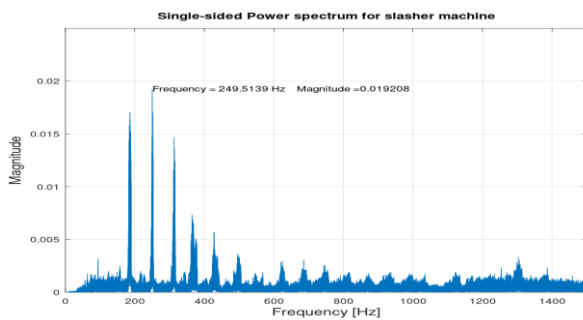


Fig 5.2 Single sided power spectrum of slasher machine

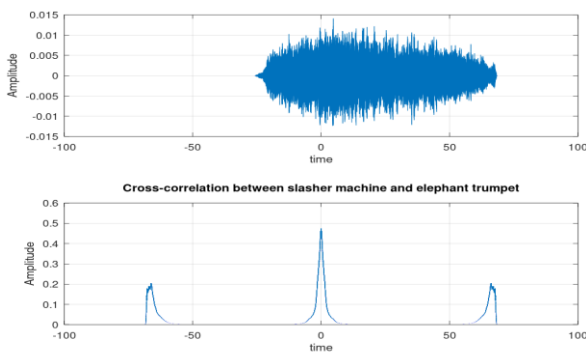


Fig 5.3 correlation between slasher machine sound and elephant sound in time domain and frequency domain respectively

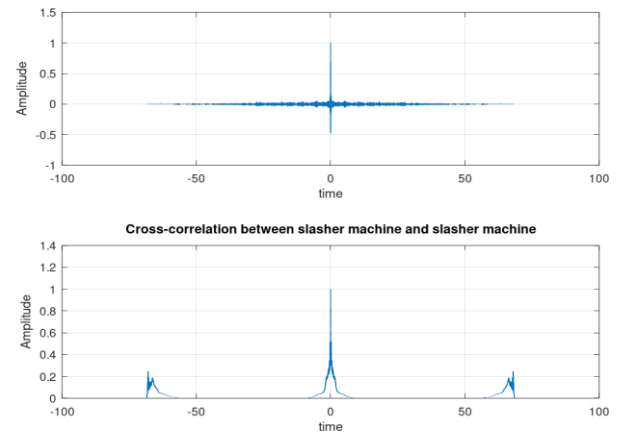


Fig 5.4 correlation between slasher machine sound and slasher machine sound in time domain and frequency domain respectively

We can see from above Fig 5.1 and 5.2, power spectrum of elephant trumpet and slasher machine are unique and has unique frequency at maximum amplitude of the signal. Plot of correlation between slasher machine sound and elephant trumpet as shown in Fig 5.3 and also correlation performed between slasher machine sound with itself in Fig 5.4. We can easily see difference in the plot when signals are matched and when signal is different.

Table 5.1: simulation result of correlation between slasher machine sound and another signal

Slasher machine sound with different signal	Correlation coefficient
Axe_chopping	0.0011512
Elephants	0.00098402
Loading on hydraulic trucks	0.0021096
Saw cutting	-0.00064524
Skidder machine	-0.0022272
Slasher machine	1
Tree breaking	0.00077336
Unloading trucks	0.0024763

Table 5.2: signal is comparison with addition of 10 percentage of random white noise to itself

Signal with signal + noise (10%)	Correlation coefficient
Axe_chopping	0.44730
Elephants	0.75815
Loading on hydraulic trucks	0.71264
Saw cutting	0.81836
Skidder machine	0.74757
Slasher machine	0.91351
Tree breaking	0.51163
Unloading trucks	0.79970

Table 5.3: signal compared with addition of 1 percentage of random white noise to itself

Signal with signal + noise (5%)	Correlation coefficient
Axe_chopping	0.70444
Elephants	0.91885
Loading on hydraulic trucks	0.89735
Saw cutting	0.94348
Skidder machine	0.91375
Slasher machine	0.97608
Tree breaking	0.76639
Unloading trucks	0.93607

Table 5.4: signal compared with addition of 5 percentage of random white noise to itself

Signal with signal + noise (1%)	Correlation coefficient
Axe_chopping	0.98035
Elephants	0.99633
Loading on hydraulic trucks	0.99520
Saw cutting	0.99754
Skidder machine	0.99607
Slasher machine	0.99901
Tree breaking	0.98618
Unloading trucks	0.99718

Coefficient of variation

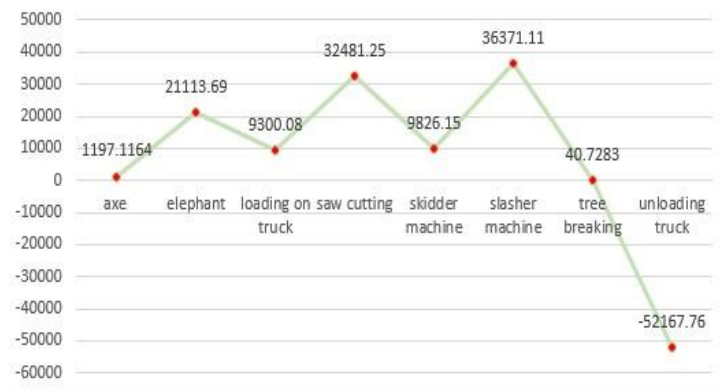


Fig 5.7: coefficient of variation of signals

Skewness

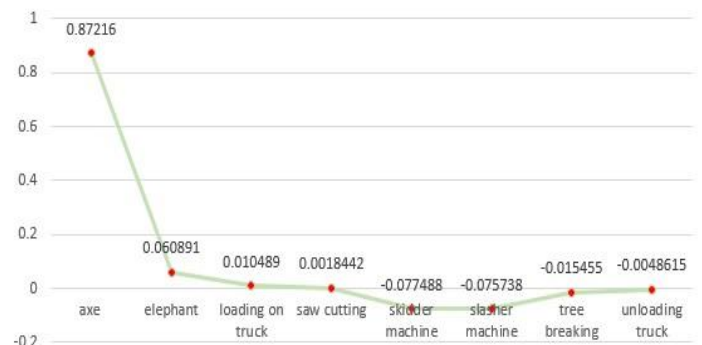


Fig. 5.9: Skewness of various signal

Kurtosis



Fig. 5.10: Kurtosis of various signal

Number of Zerocrossings



Fig. 5.5 Number of zero crossing of various signal

Standard deviation



Fig. 5.6 Standard deviation of various signals

Table 5.5 Accuracy Percentage of K- Nearest Neighbor Algorithm

Signals	2 training signals & 3 testing signals	3 training signals & 2 testing signals	4 training signals & 1 testing signal
Axe chopping	66.66	50	0
Elephant trumpet	0	100	100
Loading of trucks	100	100	100
Saw cutting	100	100	100
Skidder machine	100	100	100
Slasher machine	100	100	100
Breaking of tree	66.66	50	0
Unloading of trucks	100	100	100

Table 5.6 Confusion matrix of signals

Signal	Axe chopping	Elephant trumpet	Loading of trucks	Saw cutting	Skidder machine	Slasher machine	Breaking of tree	Unloading of trucks
Axe chopping	4	0	0	0	0	0	1	0
Elephant trumpet	0	2	0	3	0	0	0	0
Loading of trucks	0	0	5	0	0	0	0	0
Saw cutting	0	0	0	5	0	0	0	0
Skidder machine	0	0	0	0	5	0	0	0
Slasher machine	0	0	0	0	0	5	0	0
Breaking of tree	0	1	0	0	0	0	4	0
Unloading of trucks	0	0	0	0	0	0	0	5

VI. CONCLUSION

This paper discussed a model implementation of a system to detect illegal tree logging machines through extraction of data from sensors. The system was implemented with comparison of pre fed trained signals of various logging machines and comparison of these signals with test signals by adding 1% ,5% & 10% noise to the trained signals through Octave. Various signal statistics were used for comparison and K-Nearest Neighbor Algorithm was used for classification of these signals to properly identify the machine or source of noise. Further changes such as increasing test signals for identifying more diverse logging machines, and machine learning with Neural Networks can be used to determine the similar type of logging machine sounds and can be used to feed and create various types of input sounds.

VII. REFERENCES

- [1] J. Kopják and G. Sebestyén, Comparison of data collecting methods in wireless mesh sensor networks, SAMI 2018 IEEE 16th World Symposium on Applied Machine Intelligence and Informatics February 7-10 Košice, Herl'any, Slovakia
- [2] Gokay Saldamli, Sumedh Deshpande, Kaustubh Jawalekar, Pritam Gholap, Loai Tawalbeh, Levent Ertaul, Wildfire Detection using Wireless Mesh Network, 2019 Fourth International Conference on Fog and Mobile Edge Computing (FMEC)
- [3] L. K. Hema, D. Murugan and R. Mohana Priya, "Wireless sensor network-based conservation of illegal logging of forest trees," 2014 IEEE National Conference on Emerging Trends in New & Renewable Energy Sources and Energy Management (NCET NRES EM), Chennai, 2014, pp. 130-134, doi: 10.1109/NCETNRESEM.2014.7088753.
- [4] Y. Chen and J. Liaw, A novel real-time monitoring system for illegal logging events based on vibration and audio, 2017 IEEE 8th International Conference on Awareness Science and Technology (iCAST), Taichung, 2017, pp. 470-474, doi: 10.1109/ICAWS.2017.8256503.
- [5] Antonio Molina-Pico, David Cuesta-Frau, Alvaro Araujo, Javier Alejandro and Alba Rozas "Forest monitoring and Wildland early fire detection by a hierarchical wireless sensor network" Hindawi Publishing Corporation, Journal of Sensors, Volume 2016, Article ID 8325845
- [6] Junguo ZHANG, Wenbin LI, Zhongxing YIN, Shengbo LIU, Xiaolin Guo, Forest fire detection system based on wireless sensor network, 2009 4th IEEE Conference on Industrial Electronics and Applications
- [7] A. Bayo, D. Antolín, N. Medrano, B. Calvo, S. Celma, "Early detection and monitoring of forest fire with a wireless sensor network system", Proc. Eurosensors XXIV, September 5-8, 2010, Linz, Austria, Published by Elsevier
- [8] sound reference of audio signal is <https://www.zapsplat.com>