International Journal of Grid and Distributed Computing
Vol. 10, No. 11 (2017), pp.11-22
http://dx.doi.org/10.14257/ijgdc.2017.10.11.02

Screaming Sound Detection based on UBM-GMM

Suk-Hwan Jung and Yong-Joo Chung

Department of Electronics Engineering, Keimyung University
mester88@naver.com, yjjung@kmu.ac.kr

Abstract

In recent days, one of the critical social problems is that violent crimes are frequent in places like public toilets and streets. Due to this problem, the importance of surveillance systems for the safety of pedestrians is increasing gradually. As conventional visual surveillance systems have some limitations, many attempts have been made to support the system by adding audio-based functionality. In this study, we propose to use a UBM-GMM method to detect scream sounds and it is compared with conventional GMM and SVM method. From the experimental results, SVM could obtain the best False Acceptance Rate of 0.559%, which means that SVM shows a very low possibility of incorrectly deciding non-scream sound as a scream sound compared with UBM-GMM and conventional GMM. In contrary, UBM-GMM method could achieve the best average False Rejection Rate while conventional GMM could achieve the best False Rejection Rate of 12.03%, which implies that UGM-GMM and conventional GMM has relatively good sensitivity to the scream sound compared with SVM. From this study, we could conclude that both UBM-GMM (conventional GMM) and SVM have a distinctive merit from each other and further performance improvement by combining the two approaches is also expected.

Keywords: *Scream detection, surveillance system, GMM, SVM*

1. Introduction

In recent days, at places like public toilets and public parking lots as well as private houses and apartments, the rate of crime occurrences is increasing considerably [1], which makes environment security of our major concern. Surveillance systems commonly used now employ infrared sensors to detect intruders from outside or record videos as in CCTVs installed in certain places. In particular, CCTVs aid the security enforcement of the police and help prevent crimes but they suffer from environmental restrictions like vision angle and dimming light and have fatal demerit that they can't identify the crime situation in real time. To overcome this problem, there have been efforts to apply audio-based detection to the existing video surveillance system [2-4].

GMM(Gaussian Mixture Model) is one of the popular methods in audio detection as well as in speech recognition. In [5], GMMs for the door/trunk closing and car accident sound are trained respectively and they are used to detect the sound from door/trunk closing so that the black box in the car do not operate falsely by confusing it with car accident sound. In other applications, GMM has shown successful results in detecting specific sounds such as scream, shout and gun fire [6][7][8]. In [7], to detect scream and gun fire in noisy conditions, a parallel type recognizer is proposed by using the GMM for each sound.

Recently, researches using SVMs (Support Vector Machines) have been popularly used in audio detection [9][10]. SVMs are known to have equal or superior

Received (April 20, 2017), Review Result (July 21, 2017), Accepted (September 3, 2017)

ISSN: 2005-4262 IJGDC

performance in generalization compared with other classifiers. Rather than using frame-level feature vectors, they use as inputs, the means and variances of feature vectors computed periodically. In [9], the means, variances, maximum and minimum values of the 36-dimentonal MFCC(Mel-frequency cepstral coefficients) computed in every 0.25 seconds are used as the input of the SVM to show superior performance. In addition, pitch and energy values of the signal is combined with the output of the SVM to detect the existence of scream sound.

As mentioned previously, GMM and SVM are used independently in many researches to detect various audio signals but there are few research results to compare the two methods using the same audio data [12]. But it is needless to say that we need to compare them to implement more efficient classifier for scream detection. In this study, to obtain reliable classification results, we gathered various audio sound signals compared with other researches for scream detection. Also, the input feature vectors and architecture of the classifier is varied to make the comparison in more detail.

In this study, we propose to use UBM (Universal Background Model)-GMM method to make improvements to the conventional GMM. In the conventional GMM approach, independent GMMs were constructed during training for scream and non-scream data, respectively. However, in speaker recognition which has similar classification mechanism as scream detection, UBM-GMM has shown superior performance than conventional GMM [13]

The paper is organized as follows. In section II, feature extraction methods and classifiers used for scream detection are introduced. In section III, we show and compare various experimental results. Finally, in section IV, we make conclusion and discuss further studies.

2. Feature Extraction and Classifier

2.1. Feature Extraction

The waveform of sound signal can't be used as input of classifiers due to the irregularities in its characteristics. So some kind of values which can well explain the characteristics of the sound signal are used as features for the classification and traditionally, we use features like ZCR(zero crossing rate), pitch and correlation for the audio signal detection [7][9]. In this study, we use MFCC features which have shown to be quite efficient with noise-robustness in speech recognition [6].

In Figure 1, we show the process of MFCC feature extraction process in a block diagram. Audio signal sampled at 16 KHz is processed in frames each of which has 25ms duration and the interval between frames is 10ms. The audio signal is pre-emphasized as in (1) to emphasize high-frequency components of the signal and then hamming-window is used before applying FFT(Fast Fourier Transform) to the signal.

$$s(n) = s(n) - 0.9s(n-1) \tag{1}$$

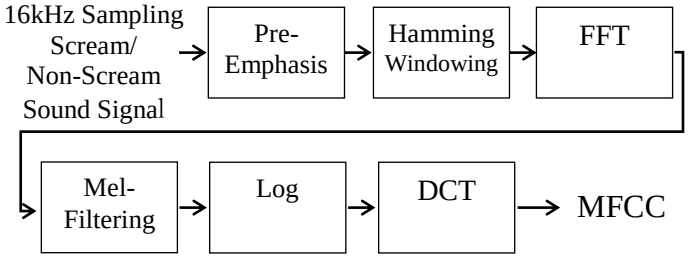

Figure 1. MFCC Feature Extraction Process

Mel-scale filter bank output is computed from the result of FFT and log of the filter bank output is transformed into 13-dimentional MFCC via DCT(Discrete Cosine Transform).

We determined the input vectors of the classifiers to give the best performance on recognition experiments. 12-dimentional MFCCs excluding c0 are used as the input vectors of the GMM while 36-dimentional MFCCs from the 12-dimentional MFCCs including both delta and acceleration coefficients are used as the input vectors of the SVM.

In this study, we used AFE(Advanced Front-End) which is defined as the standard for MFCC feature extraction by ETSI(European Telecommunication Standards Institute) [11]. AFE is expected to be robust against background noise occurring in real environments since it contains efficient algorithms to suppress noise signals.

2.2. Gaussian Mixture Model

GMM is the sum of weighted Gaussian probability density functions and has been popularly used in speech recognition to model the acoustic characteristic of speech signals in MFCC domains. Training GMM in this study is done as shown in Figure 2. After feature extraction, vector quantization is done for scream and non-scream data to find the GMM parameters for each of them. The GMM parameters consist of the weight ω_m, mean vector μ_m and covariance matrix Σ_m, $\{m = 1,2,\cdots,M\}$ of the Gaussian probability density functions comprising the GMM where M is the number of the Gaussian mixture components.

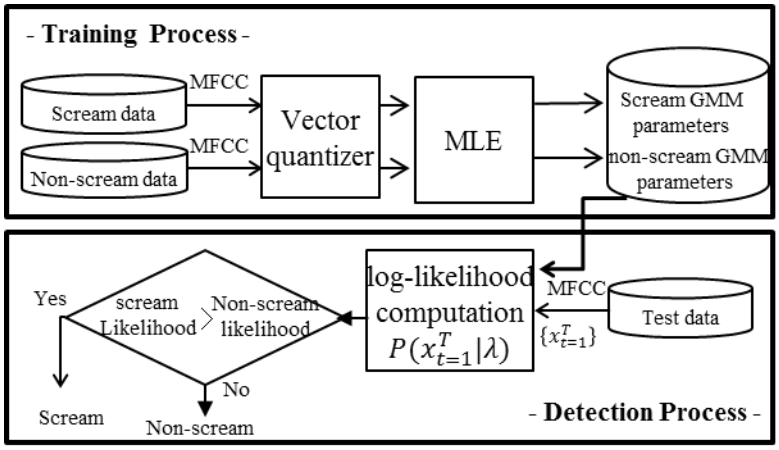

Figure 2. GMM Training and Detection Process

Given feature vectors of length T, $\{x_t\}_{t=1}^T$ the log-likelihood for each feature vector is computed as follows.

$$p(x_t) = \log\left(\sum_{m=1}^{M} \omega_m \, p_m(x_t)\right) \tag{2}$$

$$p_m(x_t) = \frac{1}{\sqrt{(2\pi)^N |\Sigma_m|}} e^{-\frac{1}{2}(x_t-\mu_m)\Sigma_m^{-1}(x_t-\mu_m)^T} \tag{3}$$

Here, $p_m(x_t)$ is the Gaussian probability density function with mean vector μ_m and diagonal covariance matrix Σ_m. For the whole feature vectors of length T, the log-

likelihood is computed by adding the log-likelihood of each feature vector by assuming that the feature vectors are independent from each other.

2.3. Universal Background Model

Universal Background Model (UBM) is a variant of the GMM methods, where a single Gaussian mixture model is first constructed using the whole training data (scream + non-scream) and adapted later using the target data (scream data). This is in contrast to the conventional GMM method where separate Gaussian mixture model is constructed for each of the scream and non-scream data. The use of UBM for scream sound detection is motivated from the idea that UBM has been successfully employed for speaker recognition which is thought to have similar classification mechanism as the scream sound detection.

Scream sound detection based on UBM can be divided into 3 steps, training, adaptation and detection process, as shown in Figure 3.

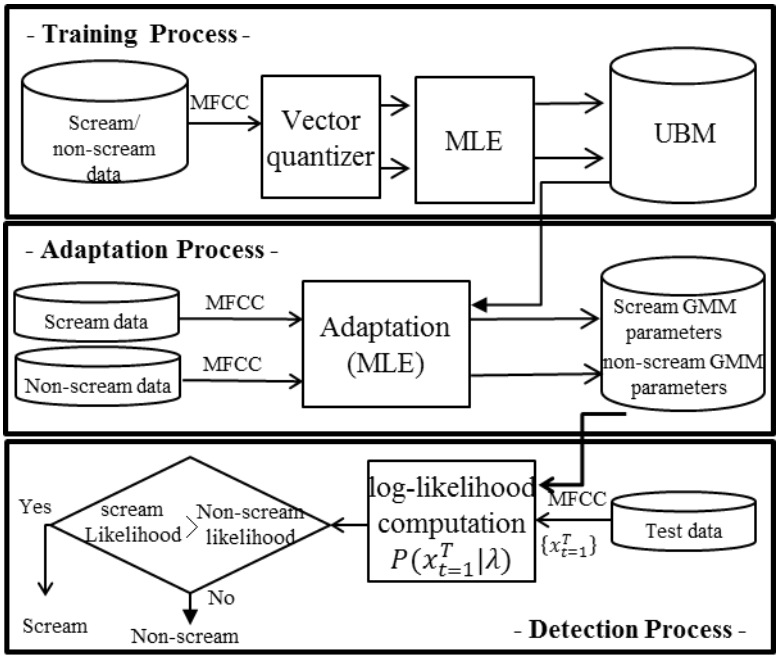

Figure 3. UBM based Scream Classification Algorithm

After UBM is constructed during training process, adaptation is done to re-estimate the UBM-GMM parameters for the scream sound. Two popular approaches are usually adopted for the UBM-GMM adaptation. One of them is Maximum A Posteriori (MAP) estimation and the other is Maximum Likelihood Estimation (MLE). In this paper, we used MLE for the adaptation since it is known that MLE performs better than MAP when there are plenty of adaptation data available.

Given MFCC feature vectors of length T, $\{x_t\}_{t=1}^{T}$ for adaptation, we can estimate the posterior probability for each of the UBM-GMM as follows using *Equation* (2).

$$P(i \mid x_t) = \frac{\omega_i p_i(x_t)}{\sum_{m=1}^{M} \omega_m p_m(x_t)} \tag{4}$$

Estimation of $P(i \mid x_t)$ in *Equation* (4) is called Expectation process in the EM(Expectation-Maximization) algorithm of the MLE adaptation. Maximization process in the EM algorithm is given in *Equations* (5) ~ (8). The MLE adaptation is done by iterating Expectation in *Equation* (4) and Maximization in *Equation* (5) ~ (8) and thereby incrementally increasing the likelihood $\Pi_{t=1}^{T} p(x_t)$.

$$n_i = \sum_{t=1}^{T} P(i \mid x_t) \tag{5}$$

$$\omega_i = n_i / \sum_{m=1}^{M} n_m \tag{6}$$

$$\hat{\mu}_i = \frac{1}{n_i} \sum_{t=1}^{T} P(i \mid x_t) x_t \tag{7}$$

$$\hat{\sigma}_i^2 = \frac{1}{n_i} \sum_{t=1}^{T} P(i \mid x_t)(x_t - \hat{\mu}_i)^2 \tag{8}$$

2.4. Support Vector Machine

SVM is a non-probabilistic binary classifier which tries to maximize the distance margin between two classes [10]. In this study, the input for the SVM is obtained by averaging the 36 dimensional MFCC feature vectors for 20 frames.

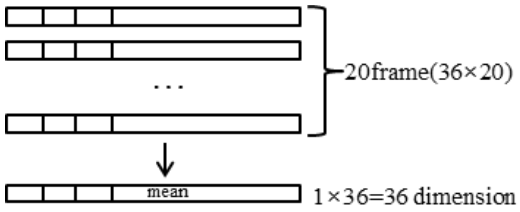

Figure 3. MFCC Input Vector Generation for SVM

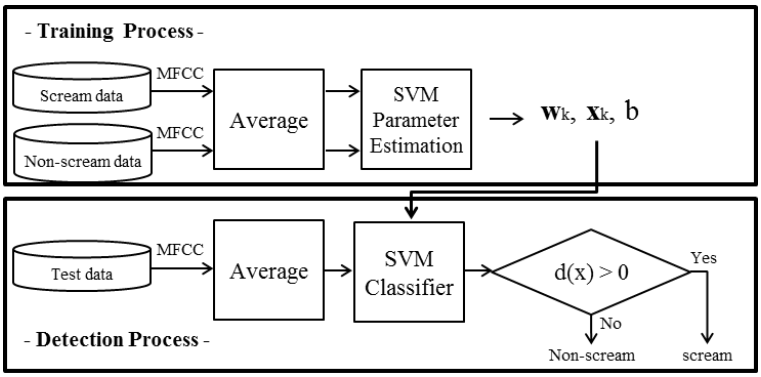

Figure 4. Training and Detection Process of SVM

In Figure 4, we show the training and detection process for the SVM in this study. During training, scream and non-scream data is inputted as in Figure 3 to estimate the SVM parameters w_k, x_k, b. During detection process, the output $d(x)$ of the SVM classifier is evaluated to determine whether the input data is scream or not.

$$d(x) = \sum_{k=1}^{K} w_k G(x_k, x) + b \quad (4)$$

Here, K is the number of support vectors. W_k, X_k, b are weights, support vectors and biases obtained during training process and G is a kernel function.

If the value of $d(x)$ is positive, we determine the input as scream sound and if negative, it is determined as non-scream sound.

3. Experiments

3.1. Database

For the experiments in this study, we used data gathered from internet [14]. The data was recorded in clean conditions. The whole data can be divided into scream data set and non-scream data set. The scream data set consists of 63 files with durations from 1 seconds to 12 seconds. The non-scream data set consists of 213 files with durations from 1 seconds to 225 seconds. For the experiments, the ratio of data amount between training and testing is set 3:1. For the reliability of the experimental results, the whole data is divided into 4 parts and Jack knife method is used for the experiments.

3.2. Experimental Results

The decision on the test data is done in every 600 ms using the labelling information. This is based on the idea that duration of the scream sound in real environments would be at least 600 ms. If the decision interval is too short, sounds like cough would be classified as scream as the sound signal is too short for us to confirm that it is not scream. In contrary, if the decision interval is too long, there is a possibility that we may miss short scream sound. To measure the performance of the classifiers, we used two popular metrics used in this field, FAR(False Acceptance Rate) and FRR(False Rejection Rate). FAR is the ratio of non-scream data which is misclassified as scream sound and FRR is the ratio of scream data which is misclassified as non-scream sound.

$$FAR = \frac{\#\ of\ decisions\ mis-classified\ as\ scream}{\#\ of\ total\ decisions\ for\ non-scream\ data}$$

$$FRR = \frac{\#\ of\ decisions\ classified\ as\ non-scream}{\#\ of\ total\ decisions\ for\ scream\ data}$$

Table 1. Experimental Results Using GMM Classifier

Mixture Number	FRR(%)	FAR(%)
1	21.14	18.64
10	22.86	15.14
20	19.74	14.17
30	21.14	13.3
40	19.74	12.78
50	**19.41**	11.85
55	20.29	11.27
60	21.43	10.67
70	22.29	10.02
76	23.43	**9.10**

In Table 1, we show the results of GMM classifier when 12-dimensional MFCCs are used as feature vectors. We can see the results depend greatly on the mixture components of the GMM. FRR has the best result of 19.41% when the number of mixture components is 50. FAR improves as the number of mixture components increases and it has the best result of 9.1% when the number of mixture components is 76. The reason for the performance improvement of FAR with the number of mixture components is that the various sound signal in the non-scream data is modelled better as the number mixture components is increased. In contrary, the improvement of FRR with the number of mixture components is limited as the acoustic characteristic of the scream sound signal is relatively stationary.

Table 2. Experimental Results Using UBM-GMM Classifier

Mixture Number	FRR(%)	FAR(%)
1	19.75	17.65
10	21.35	15.42
20	18.34	14.98
30	19.86	13.71
40	18.88	13.24
50	**18.71**	12.45
55	19.35	12.28
60	20.11	11.77
70	20.98	11.36
76	20.76	**10.21**

In Table 2, we show the results of using UBM-GMM after adapting to the scream data by MLE. Comparing with the results in Table 1, UBM-GMM shows a little performance degradation in FAR but it shows better performance in FRR. In overall, the performance of UBM-GMM does not seem to show any superiority to the conventional GMM.

In Figure 5, we show the comparison between GMM, UBM-GMM and SVM classifiers. For the GMM and UBM-GMM, the best results from Table 1 and 2 when the number of mixture components is 50(FRR) and 76(FAR) are shown. From Figure 5, we can see that SVM shows better performance than GMM and UBM-GMM in FAR (2.54% vs. 9.1% / 10.21%) while UBM-GMM is better than GMM and SVM in FRR (18.71% vs. 19.41% / 38.38%).

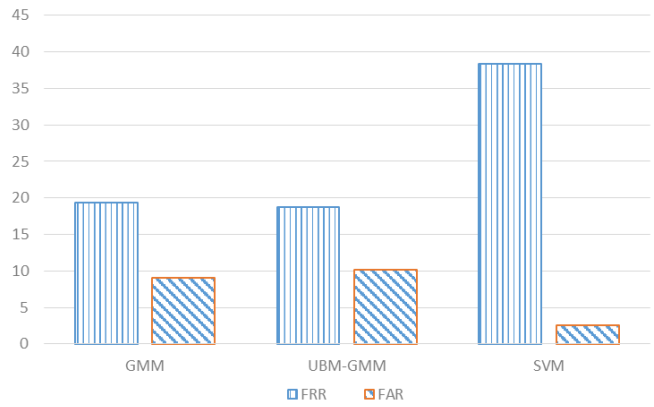

Figure 5. Performance Comparison between GMM, UBM-GMM and SVM

The scream files used in the previous experiments included background noises and breathing sound in addition to real scream sound. Thus, for the strict performance comparison, we removed those parts from the scream files and recognition experiments were done again and the results are shown in Table 3 and 4.

Table 3. Experimental Results Using GMM Classifier when Noises and Breathing Sound Signal is Removed from Scream Files

Mixture Number	FRR(%)	FAR(%)
1	**12.03**	15.43
3	26.55	11.39
5	32.78	9.15
7	34.43	7.39
10	34.85	6.01
20	38.58	5.63
30	35.68	4.76
40	36.51	4.10
50	39.83	3.58
60	43.15	2.98
70	43.15	2.61
76	44.39	**2.35**

Compared with Table 1, the results in Table 3 show much better performance in FAR but worse performance in FRR. As the characteristics of signals in a scream file is very stationary due to the removal of noises and breathing sounds, the increased number of mixture components in the GMM for scream sound seems to badly affect FRR.

In Table 4, we show the results of using UBM-GMM after adapting to the scream data by MLE. Comparing with the results in Table 3, UBM-GMM shows a little performance degradation in FAR but it shows much better performance in FRR. The detection accuracy improvement obtained in FRR overwhelms the loss in FAR. From this results, we could confirm that UBM-GMM is more effective than conventional GMM for scream sound detection.

Table 4. Experimental Results Using UBM-GMM Classifier when Noises and Breathing Sound Signal is Removed from Scream Files

Mixture Number	FRR(%)	FAR(%)
1	**13.43**	14.35
3	22.45	10.54
5	26.68	10.45
7	26.26	9.56
10	27.15	7.96
20	29.82	7.76
30	27.65	7.11
40	27.95	6.28
50	30.12	5.97
60	36.24	4.78
70	36.55	4.64
76	37.35	**4.54**

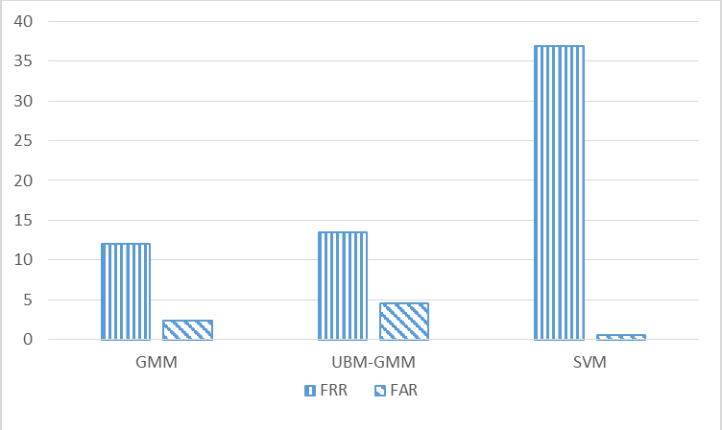

Figure 6. Performance Comparison between GMM, UBM-GMM and SVM when Noises and Breathing Sound Signal is Removed from Scream Files

In Figure 6, we compare the performance between GMM, UBM-GMM and SVM when noises and breathing sounds are removed from screaming files. For the GMM and UBM-SVM, the best results from Table 3 and 4 are occur when the number of mixture components is 1(FRR) and 76(FAR). We can see that SVM classifier has better performance than GMM and UBM-GMM in FAR (0.55% vs. 2.35% / 4.54 %) in which only a few non-scream sounds like breaking windows and cat crying are misclassified as scream sound. However, GMM shows better performance than UBM-GMM and SVM in FRR (12.03% vs. 13.43% / 36.93 %). Although the average performance of UBM-GMM is much better than GMM in FRR, the best performance is obtained by GMM when the number of mixture components is 1. This is due to the fact that the noise and breathing sound signals were removed from the original scream files to make them contain pure scream sound and as a result, no more than 1 mixture component seems to be needed to model the scream sound. But, in real situation, we may need more than one mixture component to model the scream sound and the performance of UBM-GMM is expected to be better than GMM in that case.

4. Conclusion

Surveillance systems relying on visual information do not show satisfying results due to restrictions in real environments. To overcome this problem, various audio detection methods have been proposed recently.

In this study, we proposed a UBM-GMM method and compared its performance with SVM and conventional GMM classifiers which are representative methods in audio detection. From the experiments, we could find that the methods show contrary recognition results. UBM-GMM(conventional GMM) is superior in FRR while SVM is better than UBM-GMM(conventional GMM) in FAR.

We think that the contrary results can be utilized appropriately in commercial products. For example, when we need to know exactly the moment of screaming as in crime investigation, we should not miss the screaming event and then UBM-GMM(conventional GMM) is more advantageous as they have better FRR performance. But in real time surveillance system, SVM which has better FAR is more advantageous as the system reliability is more important.

For further studies, we aim to implement a commercial product utilizing the methods mentioned in this paper. Also, a deep neural network based detection architecture will be studied for better performance. Finally, since SVM and UBM-GMM(conventional GMM) show quite different characteristics in detection results,

we will try to find a method to combine them and further improve the performance in scream sound detection.

Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by Ministry of Education(No. 2015R1D1A1A01059925). This paper is a revised and expanded version of a paper entitled [Performance Comparison between GMM and SVM for Scream Sound Detection] presented at [NGCIT 2017, Ho Chi Minh City and August 16-18, 2017].

References

[1] W. Huang, T. K. Chiew, H. Li, T. S. Kok and J. Biswas, "Scream Detection for Home Applications", Proceedings of the 5th IEEE Conference on Industrial Electronics and Applications, Taichung, Taiwan, **(2010)**, pp. 2115-2120.
[2] M. K. Nandwana, A. Ziaei and J. H. L. Hansen, "Robust Unsupervised Detection of Human Screams In Noisy Acoustic Environments", Proceedings of the International Conference on Acoustics, Speech and Signal Processing, Brisbane, Australia, **(2015)**, pp. 161-165.
[3] J. Pohjalainen, P. Alku and T. Kinnunen, "Shout Detection in Noise", Proceedings of the International Conference on Acoustics, Speech and Signal Processing, Prague, Czech Republic, **(2011)**, pp. 4968-4971.
[4] L. Gerosa, G. Valenzise, M. Tagliasacchi, F. Antonacci and A. Sarti, "Scream and Gunshot Detection in Noisy Environments", Proceedings of the 15th European Conference on Signal Processing Conference, Poznan, Poland, **(2007)**, pp. 1216-1220.
[5] B. Lei and M. W. Mak, "Sound-event Partitioning and Feature Normalization for Robust Sound-event Detection", Proceedings of the 19th International Conference on Digital Signal Processing, **(2014)**, pp. 389-394.
[6] S. G. Oh, J. U. Lee, H. S. Lee, Y. W. Chung and D. H. Park, "Abnormal Sound Detection and Identification in Surveillance System", Journal of KIISE, vol. 39, no. 2, **(2012)**, pp. 144-152.
[7] J. H. Park, J. Y. Lim, J. Y. Yang, J. M. Kyung and M. S. Hahn, "False Positive Movie Clip Decision in Black-box Using Car Door-Closing Sound Classification", Journal of IEIE, vol. 37, no. 1, **(2014)**, pp. 761-763.
[8] J. H. Seo, H. I. Lee and S. P. Lee, "A Design of a Scream Detecting Engine for Surveillance Systems", Journal of KIEE, vol. 63, no. 11, **(2014)**, pp. 1559-1563.
[9] S. Ntalampiras, I. Potamitis and N. Fakotakis, "On Acoustic Surveillance of Hazardous Situations", Proceedings of International Conference on Acoustics, Speech and Signal Processing, Taipei, Taiwan, **(2009)**, pp. 165-168.
[10] Support Vector Machines for Binary Classification, http://kr.mathworks.com/help/stats/support-vector-machines-for-binary-classification.html?requestedDomain=www.mathworks.com. [Accessed: 02, September, 2016]
[11] ETSI Draft Standard Document, Speech Processing, Transmission, and Quality Aspects (STQ); Distributed Speech Recognition; Front-end Feature Extraction Algorithm; Compression Algorithm. ETSI Standard ES 202 050, **(2002**)
[12] S. H. Jung and Y. J. Chung, "Performance Comparison between GMM and SVM for Scream Sound Detection", Proceedings of International Conference on Next Generation Computer and Information Technology, Ho Chi Minh City, Vietnam, **(2017)**, pp. 146-149.
[13] D. Reynolds, T. Quatieri and R. Dunn, "Speaker Verification Using Adapted Gaussian Mixture Model", Digital Signal Processing, vol. 10, **(2000)**, pp. 19-41.
[14] Sound Effects Download, http://www.soundsnap.com/ [Accessed: 05, September, 2016].

Authors

Suk-Hwan Jung, he received B.Sc. degrees from the Electronics, Department, Keimyung University, Daegu, South Korea, in 2017. He is now a candidate for a M.Sc. degree in Electronics Engineering at Keimyung University. His research areas include machine learning, speech recognition and their implementation with software programming.

Yong-Joo Chung, he received the B.Sc. degree in electronics engineering from Seoul National University, Seoul, South Korea in 1988 and the M.Sc. and Ph. D degrees from the Electrical and Electronics Department, Korea Advanced Institute of Science Technology, Daejon, South Korea in 1990 and 1995, respectively. He is currently a professor at the Department of Electronics at Keimyung University, Daegu, South Korea.