International Journal of Multimedia and Ubiquitous Engineering
Vol. 10, No. 1 (2015), pp. 429-436
http://dx.doi.org/10.14257/ijmue.2015.10.1.39

GA-Based Adaptive Window Length Estimation for Highly Accurate Audio Segmentation

Myeongsu Kang[1] and Jong-Myon Kim[2*]

Department of Electrical, Electronic and Computer Engineering, University of Ulsan, De 93 Daehak –ro, Nam-gu, Ulsan 680749, Korea
{ilmareboy, jongmyon.kim}@gmail.com

Abstract

Accurate audio segmentation has recently received increasing attention for its applications in automatic indexing, content analysis and information retrieval. Hence, this paper proposes a highly accurate audio segmentation methodology using a genetic algorithm-based approach to adapting and optimizing segmentation window lengths. Specifically, this paper analyzes the parameter sequence of the root-mean-square values of an input audio stream with optimal sliding window (or segmentation window) lengths found and adapted by a genetic algorithm. In addition, this paper determines whether an audio-cut occurs or not by utilizing the parameter sequences as inputs of a support vector machine. Experimental results indicate that the proposed approach achieves 100.00% and 98.69% in the average precision and recall rates of segmentation performance, respectively.

Keywords: *Audio segmentation, genetic algorithm, support vector machine, parameter sequence*

1. Introduction

Real-time multimedia services can be provided to a great number of people at once with nearly no restrictions of time and location due to the rapid advances of information technologies and multimedia devices [1-3]. However, this also causes people to be faced with information overload in their daily lives [4]. Hence, there has been an increasing demand for technologies that efficiently, quickly, and accurately browse and search for what people want from immense multimedia databases [5, 6]. Since typical multimedia databases often contain large numbers of audio signals, automatic audio retrieval for efficient production and management is significant. According to [7], there are two essential phases for reliable audio content analysis: 1) an arbitrary audio stream is first divided into a set of segments, and 2) the segments are classified into different audio classes, such as speech, music, silence, and environmental sounds. To partition audio streams, it is necessary to detect audio-cuts (or segment boundaries) that indicate abrupt changes in the audio stream [8], and the efficacy of audio content analysis highly depends on the result of audio-cut detection since it is the fundamental step of the whole audio retrieval process. Specifically, the subsequent classification process depends on the quality of the segments obtained and consequently this paper investigates a new methodology for accurate audio-cut detection.

Conventional audio segmentation methodologies are based on threshold processing of audio features such as zero-crossing rate and energy to detect audio-cuts [9, 10]. However, the accuracy of these audio segmentation processes is decreased since audio signals are recorded in noisy environments. Despite the fact that many researchers have proposed methods to calculate effective features in sliding windows [11, 12] in order to

[*] Corresponding author.

ISSN: 1975-0080 IJMUE

overcome this drawback and have shown good segmentation performance, there is no general consensus regarding window lengths that yield good features for audio segmentation. Thus, this paper proposes an adaptive segmentation window length estimation methodology by utilizing a genetic algorithm (GA) and employs a support vector machine (SVM) as an abrupt audio change detector on the resulting parameter sequence.

The rest of this paper is organized as follows. Section 2 introduces background information such as the support vector machine and genetic algorithm. Section 3 presents the proposed audio segmentation methodology using adaptive window lengths. Section 4 analyzes experimental results, and Section 5 concludes this paper.

2. Background Information

2.1. Support Vector Machine (SVM)

Support vector machine (SVM) is a supervised learning technique to analyze data and recognize patterns [13, 14]. The standard SVM is a non-probabilistic binary classifier. This paper employs an SVM to partition an audio signal into two classes, audio-cuts and non-audio-cuts. To apply the SVM to audio segmentation, this paper utilizes the Gaussian radial kernel function to map the input vector to a high-dimensional feature space because SVM performs better when used with the Gaussian radial basis kernel than with other kernels [15]. The Gaussian radial basis kernel is computed as follows:

$$k(x, y) = \exp\left(-\frac{\|x - y\|^2}{2\sigma^2}\right), \tag{1}$$

where *k(x, y)* is the kernel function, *x* and *y* are the input vectors, and σ is a parameter set by the user that determines the width of the effective basis kernel function. If the σ values chosen are too small, overtraining occurs and the basis function will be wrapped tightly around the data points. In contrast, if the σ values chosen are too large, the basis function draws an oval around the points without defining the pattern shape [15].

As described further in Section 3, the SVM, used with the Gaussian radial basis kernel, is used to detect abrupt audio changes. It takes as input a parameter sequence derived from the audio stream, the quality of whose features depend on finding optimal size segmentation window lengths. A genetic algorithm approach (introduced next) is used to find these optimal window lengths.

2.2. Genetic Algorithm (GA)

A genetic algorithm (GA) works within populations of individuals. Each individual represents a possible solution to a given problem and is assigned a fitness score according to the objective function to indicate how good a solution to the problem it is. Highly fit individuals are given opportunities to reproduce by cross breeding with other individuals in the population. This generates new individuals as offspring, which share some features with each parent. The least fit members of the population are less likely to be selected for reproduction. A new population of possible solutions is thus produced by selecting the best individuals from the current generation and mating them to generate a new set of individuals. By favoring the mating of more fit individuals, the most promising areas of the search space are explored and the population converges upon an optimal solution to the problem. This paper uses a genetic algorithm to find optimal sliding segmentation window lengths as described further in Section 3.

3. Proposed Audio Segmentation Methodology

To detect accurate audio-cuts, the proposed method first computes the root-mean-square (RMS) of the input audio stream as follows:

$$R(n) = \sqrt{\frac{1}{W_1}\sum_{m=0}^{W_1-1}\left[x(m)\right]^2}, \tag{2}$$

where *x(m)* is the value of the m^{th} sample in a sliding processing window of length W_1 in a frame in which the rectangular sliding window is used. In addition, this method calculates the parameter sequence, *P(n)*, using the RMS sequence, *R(n)*, as follows:

$$P(n) = \frac{\sum_{m=0}^{W_2-1} R(n+m)\cdot R(n+m-W_2-1)}{\sqrt{\sum_{m=0}^{W_2-1} R(n+m)^2}\cdot\sqrt{\sum_{m=0}^{W_2-1} R(n+m-W_2-1)^2}}. \tag{3}$$

An example is illustrated in Figure 1(a). The audio-cut can be detected by observing the parameter sequence *P(n),* which is defined as an autocorrelation of the root-mean-square of the signal. The sequence, *P(n)*, is calculated by using RMS values in two adjacent, nonoverlapping sliding windows with the length of W_2 as depicted in Figures 1(b) and (c).

Figure 1(b) illustrates an example of non-audio-cuts within windows L_1 and R_1. The RMS values in both windows do not abruptly change; consequently, the numerator of *P(n)* is close to its denominator and the value of *P(n)* is close to one. In contrast, Figure 1(c) shows an example of an audio-cut within window L_2. The RMS values in window L_2 abruptly change at the audio-cut, while the RMS values in window R_2 do not abruptly change. Thus, the numerator of *P(n)* is much smaller than its denominator in the window L_2, and the sequence *P(n)* is close to zero. This example illustrates how audio-cuts can be identified by observing the sequence, *P(n)*.

As a last step, the proposed method detects audio-cuts by applying SVM to the parameter sequence. To partition the input audio stream into two classes (*e.g.*, audio-cuts or non-audio-cuts), this study defines the sequence *S(n)*, which is used as an input of SVM:

$$S(n) = \left[P(n), P(n+1), \ldots, P(n+W_2-1)\right]. \tag{4}$$

In general, audio segmentation is a frame-based process and two sliding windows (W_1 and W_2) are used to differentiate between audio-cuts and non-audio-cuts. Since the accuracy of audio-cut detection depends on the size of these windows, this paper explores an adaptive approach to finding optimal window lengths by applying a genetic algorithm.

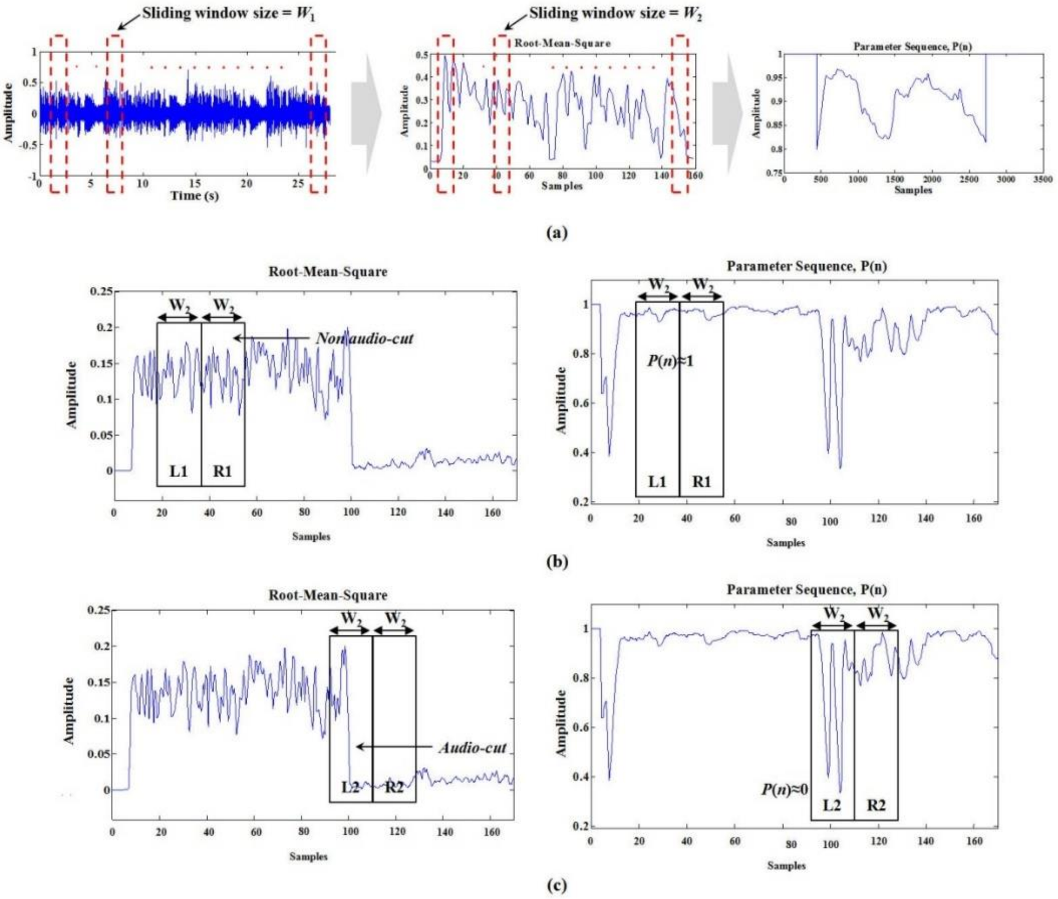

Figure 1. Audio-cut Detection using the Parameter Sequence, *P(n)*. (a) Computation Process of the Parameter Sequence. (b) An Example of non-audio-cuts using the Parameter Sequence. (c) An Example of Audio-Cuts using the Parameter Sequence

Figure 2 depicts the GA-based process used to calculate optimal window lengths. The initial population consists of randomly produced initial individuals (or chromosomes). Each chromosome is encoded in an integer form that is constructed from window lengths of W_1 and W_2 (*i.e.*, each individual has a two-dimensional point). In this process, the fitness function is defined as equation (5) to achieve the highest performance for audio segmentation:

$$P_{rate} = \frac{N_{adac}}{N_{mac}} \times 100 \ (\%), \tag{5}$$

where N_{mac} is the number of manual audio-cuts and N_{adac} is the number of all detected audio-cuts using the proposed methodology. Since the GA-based process has many parameters and mainly affects convergence speed when the population and generation numbers are too large, this study defines the population size and the generation number as 100 and 200, respectively, for experiments. In addition, this study defines crossover and mutation probabilities as 0.95 and 0.005, respectively. The iteration is stopped when the maximum generation is reached or the deviation of the fitness function is below 0.001.

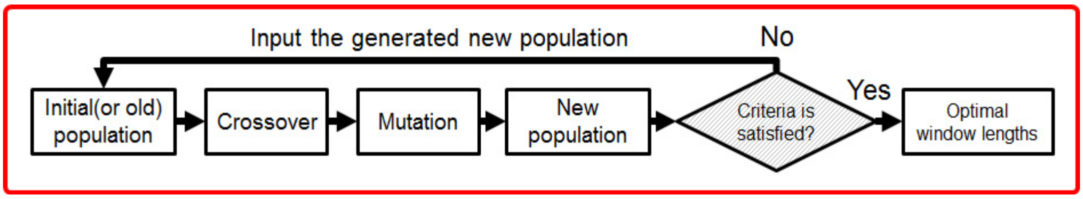

Figure 2. The Process of Exploring Optimal Window Lengths using a Genetic Algorithm

4. Experimental Results

For audio segmentation and classification simulation, this study uses four datasets composed of Korean news broadcasts obtained from Ulsan Broadcasting Corporation (http://www.ubc.co.kr). Three datasets are used for testing and one dataset was used for training. Figure 3 contrasts the audio-cuts generated based on (a) fixed window lengths with those based on (b) optimal window lengths using a GA. In Figure 3(a), some audio-cuts are missed, while in Figure 3(b), they are correctly detected. For example, in Figure 3(a), the audio-cut at time 54 occurs in the middle of a period of silence, while in Figure 3(b), the input audio stream is correctly segmented at time points 52 and 56, the beginning and end of silence.

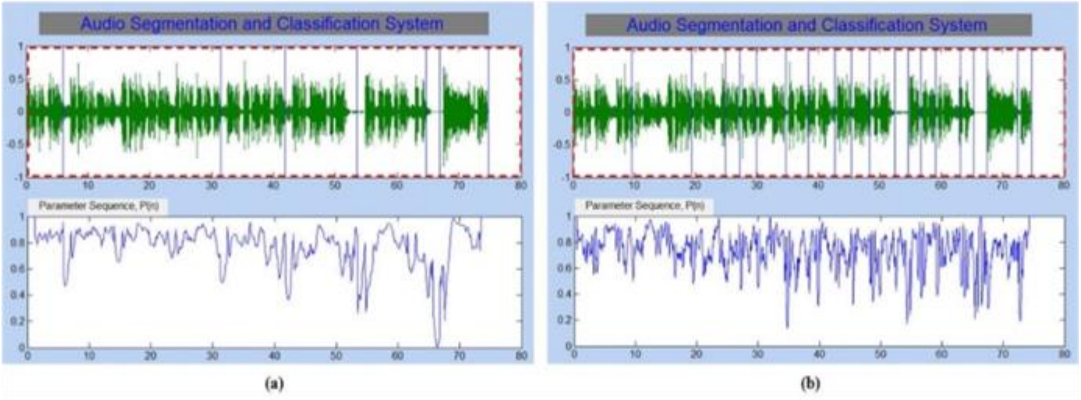

Figure 3. Audio-cut Detection. (a) Audio-cuts with the Fixed Window Lengths. (b) Audio-cuts with Optimal Window Lengths based on a Genetic Algorithm

The proposed segmentation methodology utilizing the parameter sequence with optimal window lengths results in a greater number of audio-cuts similar to real audio-cuts. However, this can lower the precision. We can have a greater number of segmented data to be classified, in which case, the segmented data do not include sufficient information to be accurately classified, and therefore a lower classification performance can result.

To address this drawback, this study further analyzed news data and found that most categories are changed at the beginning and end of periods of silence. Hence, this paper deals with the above issue by employing an additional step to remove unnecessary audio-cuts as follows:

- ***Step* 1**: The proposed method initially calculates the summation of square values for the samples (or power) in a processing window with the length of W_c. For example, in Figure 4, there are n segmented data derived from a window length W_c that is 0.13 seconds long.

- ***Step* 2**: The proposed method then computes the mean of powers and determines whether the mean value is less than a predefined threshold value *Th* to decide if the segmented data represents silence or not..This study sets *Th* as a mean value of powers for silence data in the training dataset. If each mean value of powers is less than *Th*, the proposed method marks *SEG*(*i*) equals to 0. Otherwise, the proposed method marks *SEG*(*i*) as equal to 1 until all segmented data are marked as 0 or 1.
- ***Step* 3**: If *SEG*(*i*) equals to *SEG*(*i*+1), then we remove an audio-cut at the position of the (*i*+1)th segmented data, *i*=1,2, …, *N*, where *N* is the number of all segmented data.

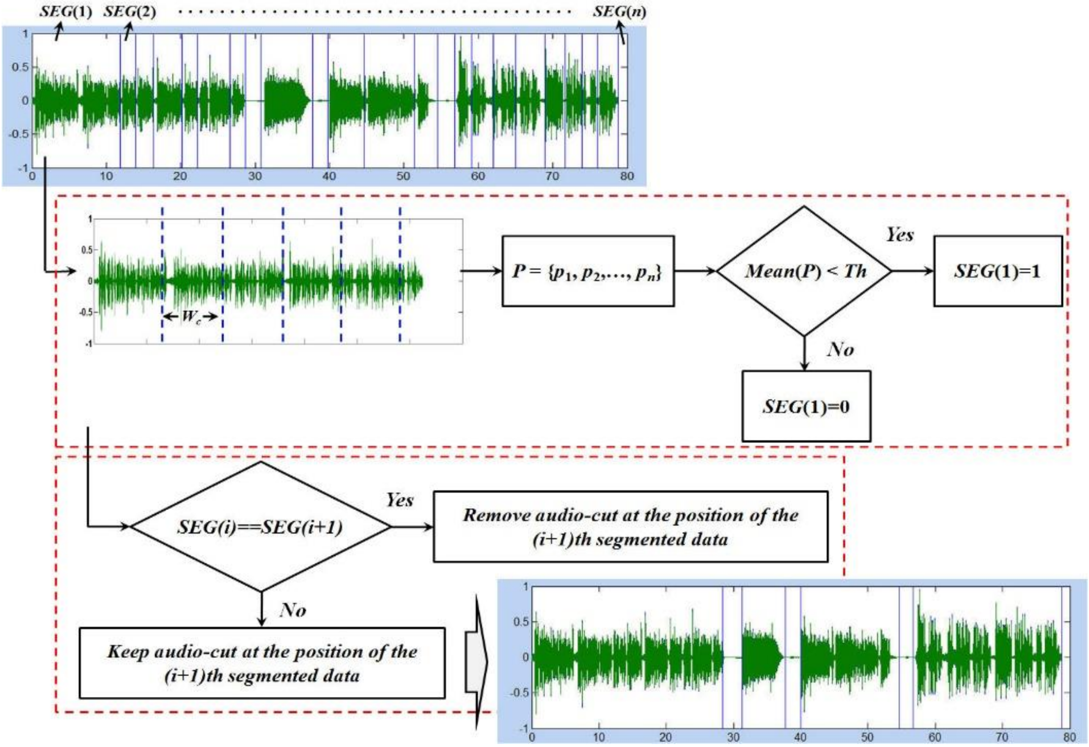
SEG(1)
SEG(2)
SEG(n)
W_c
P = {p_1, p_2,…, p_n}
Mean(P) < Th
Yes
No
SEG(1)=1
SEG(1)=0
SEG(i)==SEG(i+1)
Yes
No
Remove audio-cut at the position of the (i+1)th segmented data
Keep audio-cut at the position of the (i+1)th segmented data

Figure 4. The Process to Reduce Unnecessary Audio-Cuts

In addition, we consider segmented data as silence candidates when the length of the segmented data is longer than two seconds. Furthermore, we allow an error tolerance of 0.5 seconds to detect audio-cuts in this study.

Table 1 shows the segmentation result using the proposed segmentation methodology in terms of precision and recall rates using (5) and (6), respectively.

$$R_{rate} = \frac{N_{cdac}}{N_{mac}} \times 100 \ (\%), \tag{6}$$

where N_{cdac} is the number of correctly detected audio-cuts with ± 0.5 seconds error tolerance using the proposed approach.

Table 1. Segmentation Results of the Proposed Method in the Precision (P_{rate}) and Recall (R_{rate}) Rates

	N_{mac}	N_{adac}	N_{cdac}	P_{rate} (%)	R_{rate} (%)
Dataset 1	41	41	41	100.00	100.00
Dataset 2	50	50	49	100.00	98.00
Dataset 3	52	52	51	100.00	98.08

As presented in Table 1, the proposed method yields high segmentation performance in terms of recall and precision rates; averages of the precision rate and recall rate are 100.00% and 98.69%, respectively. In Figure 4, we decide to keep or remove audio-cuts based on the detection of silence using the mean values of power for each segmented data by comparing each mean value with a predefined threshold value. In the case of datasets 2 and 3, the segmentation performance is slightly worse than that of the dataset 1 because some speech or speech with noise signals have lower power values than a predefined *Th* value. This problem can be solved by using a more sophisticated *Th*. More details about segmentation results are available at http://eucs.ulsan.ac.kr/MDPI/information/2014-06 (*e.g.*, news data, test data, training data, segmentation results, and variable window lengths).

5. Conclusions

This paper proposes a reliable audio segmentation methodology to address the rising demand for efficient multimedia content management. To achieve higher segmentation performance, the proposed method computes root-mean-square values of the input audio stream and a parameter sequence of the root-mean-square values. In addition, this paper investigates the impact of sliding windows and uses a GA to identify optimal window lengths that yield the best segmentation performance. Moreover, the proposed method utilizes the parameter sequence as an input to a SVM to accurately determine where audio-cuts occur in the input audio stream. Furthermore, this paper evaluates segmentation performance in terms of recall and precision rates. Experimental results on three datasets from actual news broadcasts demonstrate that the proposed segmentation methodology achieves excellent segmentation performance; the average precision and recall rates are 100.00% and 98.69%, respectively.

Acknowledgements

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology(NRF-2012R1A1A2043644), and this research was supported by the Leading Industry Development for Economic Region (LeadER) grant funded the MOTIE (Ministry of Trade, Industry and Energy), Korea (No. R0001220).

References

[1] J.C. Tsai, N.Y. Yen, “Cloud-Empowered Multimedia Service: An Automatic Video Storytelling Tool,” Journal of Convergence, 4(3), **(2013)**, pp. 13-19.

[2] H. Luo and M. Shyu, “Quality of Service Provision in Mobile Multimedia - A Survey,” Human-Centric Comput. Inf. Sci., 1(5), **(2011)**, pp. 1-15.

[3] Y.S. Ho, “Challenging Technical Issues of 3D Video Processing,” Journal of Convergence, 4(1), **(2013)**, pp. 1-6.

[4] R. Shtykh and Q. Jin, “A Human-Centric Integrated Approach to Web Information Search and Sharing,” Human-Centric Comput. Inf. Sci., 1(2), **(2011)**, pp. 1-37.

[5] S. Vipparthi and S. Nagar, "Color Directional Local Quinary Patterns for Content Based Indexing and Retrieval," Human-Centric Comput. Inf. Sci., 4(6), **(2014)**, pp. 1-13.

[6] S.S. Lee, M. Shishibori, and C.Y. Han, "An Improvement Video Search Method for VP-Tree by Using a Trigonometric Inequality," Journal of Information Processing System, 9(2), **(2013)**, pp. 315-332.

[7] J. Foote, "An Overview of Audio Information Retrieval," Multimedia Systems, 7, **(1999)**, pp. 2-10.

[8] M. Davy and S. Godsill, "Detection of Abrupt Spectral Changes Using Support Vector Machines: An Application to Audio Signal Segmentation," In Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, **(2002)**, pp. 1313-1316, Florida, USA.

[9] T. Giannakopoulos, A. Pikrakis, and S. Theodoridis, "A Novel Efficient Approach for Audio Segmentation," In Proceedings of the 19th International Conference on Pattern Recognition, **(2008)**, pp. 1-4, Florida, USA.

[10] N. Nitanda, M. Haseyama, and H. Kitajima, "Audio Signal Segmentation and Classification for Scene-Cut Detection," In Proceedings of IEEE International Symposium on Circuits and Systems, **(2005)**, pp. 4030-4033, Kobe, Japan.

[11] M. Bartsch and G. Wakefield, "Audio Thumbnailing of Popular Music Using Chroma-Based Representations," IEEE Trans. Multimedia, 7(1), **(2005)**, pp. 96-104.

[12] K. West and S. Cox, "Finding an Optimal Segmentation for Audio Genre Classification," **(2005)**, Queen Mary, University of London.

[13] G.M. Nagi, R. Rahmat, F. Khalid, and M. Taufik, "Region-Based Facial Expression Recognition in Still Images," Journal of Information Processing Systems, 9(1), **(2013)**, pp. 173-188.

[14] Y.S. Hwang, J.B. Kwon, J.C. Moon, and S.J. Cho, "Classifying Malicious Web Pages by Using an Adaptive Support Vector Machine," Journal of Information Processing Systems, 9(1), **(2013)**, pp. 395-404.

[15] S.R. Gunn, "Support Vector Machines for Classification and Regression," **(2014)**, Technical Report.

Authors

Myeongsu Kang received BS and MS degrees in computer engineering and information technology in 2008 and 2010, respectively, from the University of Ulsan in Ulsan, South Korea, where he is a currently PhD student of electrical, electronics, and computer engineering. His current research interests include machinery fault diagnosis and condition monitoring, high-performance multimedia signal processing, and application-specific SoC design.

Jong-Myon Kim received a BS in electrical engineering from Myongji University, Yongin, Korea, in 1995, an MS in electrical and computer engineering from the University of Florida, Gainesville, in 2000, and a PhD in electrical and computer engineering from the Georgia Institute of Technology, Atlanta, in 2005. He is an associate professor of Electrical Engineering at the University of Ulsan, Korea. His research interests include multimedia processing, multimedia-specific processor architecture, parallel processing, and embedded systems. He is a member of IEEE and the IEEE Computer Society.