

특집논문 (Special Paper)

방송공학회논문지 제19권 제6호, 2014년 11월 (JBE Vol. 19, No. 6, November 2014)

<http://dx.doi.org/10.5909/JBE.2014.19.6.819>

ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

## 오디오 콘텐츠를 위한 비음수 행렬 분해 기법 기반의 실시간 단일채널 배경 잡음 추출 기법

이 석 진<sup>a)†</sup>

### Online Monaural Ambient Sound Extraction based on Nonnegative Matrix Factorization Method for Audio Contents

Seokjin Lee<sup>a)†</sup>

#### 요 약

본 논문에서는 비음수 행렬 분해 (NMF) 기법을 이용하여 단일 채널에서 배경음 성분을 추출하는 알고리즘에 대해 서술한다. 이러한 배경음 성분 추출은 오디오 업믹싱 시스템을 고려하여 개발되었으며, 기존의 연구를 통하여 분리된 배경음 신호가 업믹싱 시스템에 적용될 경우 공간감을 향상시킬 수 있다는 사실이 이미 확인된 바 있다. 다만 기존의 기법은 음향 신호를 모두 축적하여 일괄적으로 처리해야 한다는 단점이 있어, 스트리밍 시스템이나 디지털 시그널 프로세서 (DSP) 등을 이용한 시스템에서 사용되기 어렵다. 본 논문에서는 이를 해소하기 위하여 실시간 비음수 행렬 분해 기법을 이용한 배경음 추출 시스템을 고안하여 실험하였다. 실험에서 처리된 음원을 스펙트럼 평활도를 이용하여 분석한 결과, 고안된 배경음 추출 시스템이 기존의 일괄 추출 시스템과 유사한 정도로 배경음 성분을 추출했음을 확인할 수 있었다.

#### Abstract

In this paper, monaural ambient component extraction algorithm based on nonnegative matrix factorization (NMF) is described. The ambience component extraction algorithm in this paper is developed for audio upmixing system; Recent researches have shown that they can enhance listener envelopment if the extracted ambient signal is applied into the multichannel audio upmixing system. However, the conventional method stores all of the audio signal and processes all at once, so it cannot be applied to streaming system and digital signal processor (DSP) system. In this paper, the ambient component extraction algorithm based on on-line nonnegative matrix factorization is developed and evaluated to solve the problem. As a result of analysis of the processed signal with spectral flatness measures in the experiment, it was shown that the developed system can extract the ambient signal similarly with the conventional batch process system.

Keyword : Nonnegative Matrix Factorization, Online Nonnegative Matrix Factorization, Multichannel Audio Upmixing, Ambient Signal Extraction

## I. 서론

음향 녹음 및 재생 기기의 등장으로 청취자가 가정에서 오디오를 향유하게 된 이래, 음질뿐만 아니라 공간감에 대해서도 보다 충실하고 현장감이 있는 소리를 들을 수 있기를 바라는 요구가 항상 존재해 왔다. 이에 따라 과거 모노채널 오디오 시스템에서부터 시작하여, 스테레오 시스템, 그리고 5.1채널 서라운드 시스템까지 다채널 오디오 시스템이 발전하면서, 청취자에게 보다 실감나는 음향 재생 성능을 제공할 수 있게 되었다<sup>[1]</sup>.

보다 향상된 다채널 스피커 시스템의 성능을 즐기기 위해서는, 재생해야 할 음향 신호 또한 스피커 시스템에 맞도록 구성되어야 한다. 특히 최근에는 5.1채널 서라운드 이상의 스피커 시스템에 대한 연구 개발이 지속되면서, 다양한 다채널 오디오 시스템이 소개되고 있다. 이러한 상황에서 보다 적은 채널 개수의 음향 신호를 다채널 시스템에 맞도록 재구성하거나, 혹은 최근에 개발되는 다양한 오디오 시스템에 적합한 콘텐츠를 확보하기 위하여 오디오 업믹싱 기법이 개발되고 있다.

특히, 5.1채널 서라운드의 좌/우 서라운드 채널이나, 상방향 스피커가 있는 시스템에서 상방향 레이어 (top layer) 채널의 신호들은 주로 뚜렷한 음상을 맺도록 하기 보다는 넓은 음장감을 제공하는 데에 초점을 두고 있다. 이에 따라 다채널 업믹싱 기법에 있어서 주요한 역할을 하는 부분 중 하나가 배경음 분리 기법이다.

기존의 멀티채널 오디오 업믹싱 기법은 주로 채널간의 정보, 즉 믹싱 과정에 있어 음상이 맺히는 공간에 대한 정보를 이용하여 업믹싱 대상 신호를 추출하는 방법을 취하고 있다<sup>[2,3]</sup>. 그러나 이러한 기법들은 입력 신호가 단일채널인

경우, 그리고 수평 방향이 아닌 수직 방향의 업믹싱을 수행할 경우에는 적용될 수 없다.

최근 독일 Fraunhofer 연구소의 한 연구결과에 따르면, 비음수 행렬 분해 (Nonnegative Matrix Factorization: NMF) 기법을 이용하여 단일채널에서 배경음에 가까운 소리들을 분리하였으며, 해당 기법으로 분리한 배경음을 이용하여 업믹스 시스템에 적용하였을 경우 기존 기법 대비 향상된 청취평가 결과를 얻을 수 있었다<sup>[4]</sup>.

위의 연구 결과가 좋은 성능을 보여주시는 했으나, 기존 비음수 행렬 분해 기법의 경우 전체 음원을 한꺼번에 처리해야 한다는 단점을 가지고 있다. 즉, 시간 지연에 민감한 스트리밍 데이터 시스템이나, 홈시어터 플레이어와 같이 디지털 시그널 프로세서(DSP)를 사용한 시스템에서는 사용할 수 없는 알고리즘이다.

본 논문에서는 위와 같은 비음수 행렬 분해에 기반한 실시간 배경음 추출 기법에 대해 논의하며, 특히 실시간 비음수 행렬 분해에 기반한 기법을 도출한다. 이를 토대로 기존의 일괄적 비음수 행렬 분해 기법에 의해 추출된 결과와 비교 분석한 후, 실제 음악 신호를 통하여 추출된 음원을 분석하여 성능을 검증한다.

## II. 비음수 행렬 분해 기법에 기반한 배경음 추출 알고리즘

### 1. 비음수 행렬 분해 기법

비음수 행렬 분해 기법이란 다음과 같이  $K \times N$  크기의 비음수 행렬  $V$  를  $K \times R$  크기의 비음수 행렬  $W$  와  $R \times N$  크기의 행렬  $H$  의 곱으로 나타낸 후, 각 행렬을 추정하여 분해하는 기법이다<sup>[5,6]</sup>.

$$V = WH + E \tag{1}$$

일반적으로 오디오 신호처리 시스템에서 행렬  $V$  는 음향 신호의 크기 스펙트럼(magnitude spectrogram)을 사용하며, 이 경우 행렬  $W$  는 주파수 영역 기저를, 행렬  $H$  는 시간 영역 기저를 나타낸다.

a) 경기대학교 전자공학과 (Department of Electronic Engineering, Kyonggi University)

‡ Corresponding Author : Seokjin Lee

E-mail: sjlee6@kgu.ac.kr

Tel: +82-31-249-9807

※ 본 연구는 경기대학교 제조혁신특성화사업단(2014년)의 지원으로 이루어졌음.

※ 본 논문의 연구결과 중 일부는 “한국방송공학회 2014년 하계학술대회”에서 발표한 바 있음.

· Manuscript received September 21 2014 Revised November 19, 2014

Accepted November 19, 2014

위와 같은 비음수 행렬 분해 모델에서 각 기저 행렬, 즉  $\mathbf{W}$  와  $\mathbf{H}$  를 추정하는 기법에는 여러 방법이 있으나, 주로 많이 사용되는 기법은 다음과 같이 곱셈에 의한 갱신 (multiplicative update) 방법에 의해 순차적으로 추정하는 기법이다<sup>[7]</sup>.

$$\mathbf{W} \leftarrow \mathbf{W} \otimes [(\mathbf{V}\mathbf{H}^T) \oslash (\mathbf{W}\mathbf{H}\mathbf{H}^T)] \quad (2)$$

$$\mathbf{H} \leftarrow \mathbf{H} \otimes [(\mathbf{W}^T\mathbf{V}) \oslash (\mathbf{W}^T\mathbf{W}\mathbf{H})] \quad (3)$$

여기서  $\otimes$  는 Hadamard product로 행렬 원소 간의 곱셈을 의미하고,  $\oslash$  는 원소 간의 나눗셈을 의미한다.

## 2. 비음수 행렬 분해에 기반한 배경음 추출 기법

비음수 행렬 분해는 오디오 신호의 희박 표현 (Sparse Representation)에 의거하여, 해당 음악 신호를 의미를 가진 여러 개의 음표 성분으로 분리하는 방법으로 주로 사용된다. 이 과정에서 희박 성분으로 잘 표현되지 않는, 전 주파수 영역에 걸쳐 퍼져 있는 소리의 성분들은 기저 행렬로 잘 표현되지 않는 결과를 보인다. 이를 이용하여, 비음수 행렬 분해로 분석하고 남은 나머지 신호들을 따로 추출하면 배경음 성분을 추출해 낼 수 있다<sup>[4]</sup>. 단, 여기서 추출되는 배경음 성분은 기존의 공간적인 배경음과 조금 다른 성질을 가진다. 추출된 배경음 성분은 단일 채널의 주파수 구조를 통해서 얻어낸 배경음 성분이며, 따라서 공간적인 분포와는 관계없이 주파수 영역에 걸쳐 고르게 분포되는 성분들을 배경음으로 간주하여 추출해내기 때문이다.

배경음 성분을 추출해내기 위하여, 먼저 위의 비음수 행렬 분해 결과를 이용하여 배경음 성분의 크기 성분을 먼저 다음과 같이 계산한다<sup>[4]</sup>.

$$|\mathbf{A}_{kn}| = \beta_{kn} (\mathbf{V} - \mathbf{W}\mathbf{H})_{kn} \quad (4)$$

여기서  $|\mathbf{A}_{kn}|$  는 추출할 배경음 성분의  $k$  번째 주파수 빈 및  $n$  번째 시간 윈도우의 스펙트럼 크기 성분을 나타낸다. 여기서 크기 성분은 항상 양의 값을 가져야 하지만,  $\mathbf{V} - \mathbf{W}\mathbf{H}$  의 성분은 항상 양의 값을 가지지 않는다. 따라서,

양의 값을 가지도록 가중치  $\beta_{kn}$  을 다음과 같이 설정한다<sup>[4]</sup>.

$$\beta_{kn} = \begin{cases} \gamma & (\mathbf{V} - \mathbf{W}\mathbf{H})_{kn} < 0 \\ 1 & otherwise \end{cases} \quad (5)$$

여기서  $\gamma$  는 -1과 0사이의 상수 값으로 설정한다. 위와 같이 얻은 배경음 성분의 크기 성분에 더하여, 다음과 같이 입력 신호의 위상 정보를 함께 이용하여 역 푸리에 변환을 사용함으로써 추출할 배경음 성분 신호를 얻어낸다<sup>[4]</sup>.

$$\mathbf{A}_{kn} = |\mathbf{A}_{kn}| \angle \mathbf{X}_{kn} \quad (6)$$

여기서  $\angle \mathbf{X}_{kn}$  는  $k$  번째 주파수 빈 및  $n$  번째 시간 윈도우의 입력 신호의 위상을 나타낸다.

## III. 실시간 배경음 추출 기법

### 1. 실시간 비음수 행렬 분해 기법

앞서 언급된 바와 같이, 기존의 비음수 행렬 분해 기법은 모든 데이터를 축적한 뒤 일괄적으로 계산해야 한다는 단점이 있으며, 이를 극복하기 위하여 실시간으로 기저 행렬을 추정하는 알고리즘들이 개발되고 있다. 그 중 하나가 재귀적 최소 제곱법 (Recursive Least Squares: RLS)의 원리를 이용하여 개발된 실시간 비음수 행렬 기법 알고리즘이다<sup>[8,9]</sup>.

실시간 비음수 행렬 분해 기법은 매  $n$  번째 시간 윈도우마다 다음의 연산을 통해 주파수 기저 행렬 및 시간영역 기저 벡터를 추정한다. 먼저  $n$  번째 시간 윈도우의 시간영역 기저벡터는 다음과 같이 계산된다<sup>[8,9]</sup>.

$$\mathbf{h}(n) = [\mathbf{W}^\dagger(n-1)\mathbf{v}(n)]_+ \quad (7)$$

여기서  $\mathbf{h}(n)$  은  $R \times 1$  크기의 시간영역 기저 벡터이며,  $\mathbf{v}(n)$  은  $K \times 1$  크기를 가지는  $n$  번째 시간 윈도우의 크기 스펙트럼 데이터를 나타낸다. 또한  $[\ ]_+$  는 0보다 작은 값을 0이 되도록 연산하는 반파 정류 연산을 의미하며,  $\dagger$  는 다음과 같은 무어-펜로즈 의사역행렬 (Moore-Penrose pseudo inverse matrix)를 나타낸다.

$$\mathbf{W}^\dagger(n-1) = [\mathbf{W}^H(n-1)\mathbf{W}(n-1)]^{-1}\mathbf{W}^H(n-1) \quad (8)$$

위와 같이 추정된 시간영역 기저 벡터의 값을 이용하여, 주파수 기저 벡터는 다음과 같이 추정된다[8,9]. 먼저 주파수 빈에 관계없이 공통으로 계산되는 이득 벡터  $\mathbf{k}(n)$  과 시간 기저 벡터  $\mathbf{h}(n)$ 의 자기-상관 행렬 (auto-correlation matrix)의 역행렬인  $\mathbf{P}(n)$  은 다음과 같이 계산된다.

$$\mathbf{k}(n) = \frac{\mathbf{P}(n-1)\mathbf{h}(n)}{\lambda + \mathbf{h}^T(n)\mathbf{P}(n-1)\mathbf{h}(n)} \quad (9)$$

$$\mathbf{P}(n) = \lambda^{-1}\mathbf{P}(n-1) - \lambda^{-1}\mathbf{k}(n)\mathbf{h}^T(n)\mathbf{P}(n-1) \quad (10)$$

여기서 (9), (10)의 연산 결과를 이용하여 매  $k$  번째 주파수 빈 마다 다음의 연산을 통해 주파수 기저를 추정한다.

$$\xi_k(n) = v_k(n) - \mathbf{w}_k(n-1)\mathbf{h}(n) \quad (11)$$

$$\mathbf{w}_k^T(n) = [\mathbf{w}_k^T(n-1) + \mathbf{k}(n)\xi_k^T(n)]_+ \quad (12)$$

여기서  $v_k(n)$  은  $\mathbf{v}(n)$  의  $k$  번째 주파수 성분 값을 나타내고,  $\mathbf{w}_k(n)$  은  $\mathbf{W}(n)$  행렬의  $k$  번째 행 벡터를 나타낸다. 그리고 망각 인자  $\lambda$  는 일반적으로 1에 가까운 값을 가지며, 과거 값에 대한 가중치를 나타낸다.

## 2. 실시간 배경음 분리 기법

위와 같이 추정된 기저 행렬 및 벡터 값을 이용하여, 배경음 신호의 크기 성분  $a_k(n)$  은 다음과 같이 계산된다.

$$a_k(n) = \beta_k(n)(v_k(n) - \mathbf{w}_k(n)\mathbf{h}(n)) \quad (13)$$

매 시간 윈도우 결과 사이의 변화가 너무 큰 경우 시간적 불연속에 의한 잡음이 들리는 경우가 있으므로, 이를 방지하기 위하여 다음과 같이 시간 영역 평활화(smoothing) 작업을 추가로 진행한다.

$$a_k(n) \leftarrow (1-\eta)a_k(n-1) + \eta a_k(n) \quad (14)$$

기존의 배경음 추출 기법과 마찬가지로, 추출되는 배경음 신호의 위상 성분 값은 입력 신호의 위상 성분 값과 동일한 값을 사용한다.

## IV. 실험 결과 및 분석

### 1. 실험 환경 및 평가 방법

본 논문에서 소개 및 도출된 배경음 추출 기법의 동작을 분석하기 위하여, 음악 신호에 대한 배경음 추출 실험을 진행하였다. 실험 신호로는 RWC (Real World Computing) music database[10]의 “popular music” 데이터베이스 신호를 사용하였다. 해당 데이터베이스에는 총 16 트랙의 음원이 있으며, 본 실험에서는 각 음원 트랙들을 20초 간격의 세그먼트로 나누어 알고리즘을 수행하고, 성능 지표를 계산하여 이에 대한 평균을 구하였다. 스펙트럼 분석은 2048 길이의 Hamming 윈도우를 이용하여 50% 겹침(overlap)을 통하여 수행하였다. 기존의 일괄 비음수 행렬 분해 기법과 제안하는 실시간 비음수 행렬 분해 기법의 기저 개수는 32개로 설정하였으며, 실시간 비음수 행렬 분해 기법의 망각 인자  $\lambda$  는 1로, 그리고 실시간 배경음 추출 기법의 평활화 인자  $\eta$  는 0.8로 설정하였다.

앞서 언급한 바와 같이, 비음수 행렬 분해 기법 기반의 배경음 추출 기법은 스펙트럼 상에서 에너지가 균일하게 분포되어 있는 부분을 추출하는 기법이다. 따라서, 분리된 배경음 성분의 에너지가 고르게 분포되어 있는 정도를 측정함으로써 각 알고리즘이 설계된 목적대로 잘 동작하고 있는지를 평가할 수 있다.

에너지가 스펙트럼 상에서 고르게 분포되어 있는 정도를 측정하는 인자로, 음악 정보 분석 분야에서 사용되는 spectral flatness 인자(SFM)와 spectral crest 인자(SCM)가 있다. Spectral flatness 인자는 다음과 같이 계산된다<sup>[11]</sup>.

$$SFM(n) = \frac{\left( \prod_{k=0}^{K-1} |a_k(n)| \right)^{1/K}}{\frac{1}{K} \sum_{k=0}^{K-1} |a_k(n)|} \quad (15)$$

위와 같은 spectral flatness 인자는 스펙트럼 크기의 산술 평균과 기하평균의 비율을 의미한다. 해당 값은 백색잡음 일 때 가장 큰 값으로 1의 값을 가지며, 단일 톤(tone)의 신호일 때 가장 작은 값인 0의 값을 가진다.

한편, 두 번째 평가지표인 spectral crest 인자는 다음과 같이 정의된다<sup>[11]</sup>.

$$SCM(n) = \frac{\max |a_k(n)|}{\frac{1}{K} \sum_{k=0}^{K-1} |a_k(n)|} \quad (16)$$

위의 비음수 행렬 분해 기법 기반의 배경음 추출 기법들은 평활한 주파수 성분들을 추출해 내는 알고리즘이므로, 제안하는 실시간 배경음 추출 기법을 통해 얻은 신호들의 SFM과 SCM이 일괄 배경음 추출 기법의 값과 유사하다면, 실시간 배경음 추출 기법이 과거 데이터를 사용하지 않고도 일괄 배경음 추출 기법과 유사한 결과를 얻을 수 있다는 결론을 얻을 수 있다.

위와 같은 SFM 및 SCM에 의한 객관평가 외에, MUSHRA (Multiple Stimuli with Hidden Reference and Anchor) 기법을 이용한 주관청취평가도 시행되었다. 청취평가는 특별한 음악적 훈련을 받지 않은 일반인 10명을 대상으로 진행되었으며, 기존 기법인 일괄 배경음 추출 기법의 처리 음원과 제안된 실시간 배경음 추출 기법의 처리 음원의 유사도를 측정하는 형태로 진행되었다. 비교 대상은 일괄 배경음 추출 기법의 처리 결과 음원, 실시간 배경음 추출 기법의 처리 결과 음원, 1 kHz를 기준으로 하는 고역통과필터의 처리 결과 음원, 그리고 처리를 하지 않은 원본 결과 음원의 4가지 음원을 선정하였다. 일반적인 음악 음원의 경우, 배경음 성분이 주로 고역에 많아 배경음 추출 기법의 처리 결과는 주로 고주파 성분에 에너지가 더 많은 결과를 가진다. 따라서, 이에 대한 비교 음원으로 위와 같이 1 kHz를 기준으로 하는 고역통과필터의 처리 음원을 사용하였다. 또한, 여기서 원본 결과 음원은 최저 기준(anchor)로써 제시되었다. 위와 같이 1개의 음원 트랙 당 20초 길이의 음원 4개를 비교하여 점수를 부여하도록 하였으며, 위와 같은 실험을 총 16개 트랙에 걸쳐 진행하였다.

## 2. 실험 결과 및 분석

표 1에 위의 실험 결과가 도시되어 있다. 표 1에 도시되어 있는 실험 결과는 각 트랙에 포함되어 있는 세그먼트 결과의 평균을 나타낸다. 첫째 열은 입력 신호의 SFM과 SCM 값을 나타내고 있으며, 이는 성능 비교를 위한 최저기준값(anchor)으로 사용되기 위해 도시되었다. 둘째 열은 기존 일괄 추출 알고리즘의 분석 결과를, 그리고 셋째 열은 제안하는 실시간 추출 알고리즘의 결과를 보이고 있다.

표 1. 실험 결과  
 Table 1. Results of the experiment

|          | Input Signal |      | Batch ext. |      | On-line ext. |      |
|----------|--------------|------|------------|------|--------------|------|
|          | SFM          | SCM  | SFM        | SCM  | SFM          | SCM  |
| Track 1  | 0.27         | 0.04 | 0.39       | 0.19 | 0.42         | 0.17 |
| Track 2  | 0.26         | 0.05 | 0.40       | 0.17 | 0.39         | 0.16 |
| Track 3  | 0.28         | 0.06 | 0.41       | 0.18 | 0.41         | 0.16 |
| Track 4  | 0.26         | 0.04 | 0.45       | 0.17 | 0.44         | 0.15 |
| Track 5  | 0.27         | 0.06 | 0.38       | 0.18 | 0.41         | 0.18 |
| Track 6  | 0.29         | 0.06 | 0.38       | 0.18 | 0.40         | 0.18 |
| Track 7  | 0.29         | 0.06 | 0.42       | 0.18 | 0.41         | 0.17 |
| Track 8  | 0.21         | 0.05 | 0.35       | 0.16 | 0.38         | 0.14 |
| Track 9  | 0.24         | 0.07 | 0.36       | 0.17 | 0.38         | 0.17 |
| Track 10 | 0.30         | 0.07 | 0.40       | 0.18 | 0.42         | 0.17 |
| Track 11 | 0.18         | 0.06 | 0.37       | 0.16 | 0.37         | 0.14 |
| Track 12 | 0.21         | 0.06 | 0.36       | 0.16 | 0.37         | 0.16 |
| Track 13 | 0.20         | 0.07 | 0.32       | 0.18 | 0.32         | 0.16 |
| Track 14 | 0.31         | 0.06 | 0.42       | 0.21 | 0.43         | 0.21 |
| Track 15 | 0.30         | 0.05 | 0.41       | 0.20 | 0.43         | 0.19 |
| Track 16 | 0.28         | 0.06 | 0.38       | 0.18 | 0.39         | 0.16 |
| Average  | 0.26         | 0.06 | 0.39       | 0.18 | 0.40         | 0.17 |

기존 일괄 추출 알고리즘 결과와 제안하는 실시간 추출 알고리즘 결과를 비교하여 보면, 일괄 추출 알고리즘의 SFM은 0.32~0.45, SCM은 0.16~0.21의 분포를 보임을 확인할 수 있으며, 실시간 추출 알고리즘 결과의 SFM은 0.32~0.44, SCM은 0.14~0.21의 분포를 보인다. 이는 입력 신호의 SFM이 0.18~0.31, SCM이 0.04~0.07의 분포를 가지는 것과 비교하여 볼 때, 두 결과가 상대적으로 유사한 수치를 보임을 확인할 수 있다. 평균 값 또한 일괄 추출 알고리즘과 실시간 추출 알고리즘이 SFM, SCM 모두 0.01가

량의 작은 차이를 보이는 것을 확인할 수 있다. 그림 1에 도시된 알고리즘 별 평균 및 표준편차의 그래프를 보면 위의 사실을 보다 쉽게 확인할 수 있다.

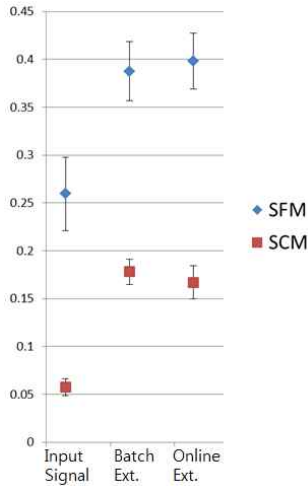


그림 1. 실험 결과의 평균 및 표준편차  
Fig. 1. Averages and standard deviations of the experimental results

특히, 트랙 별 성능을 비교하여 보면 성능 평가 지표가 최대 0.03 정도 차이가 보이는 것을 확인할 수 있으며, 입력 신호의 값과 추출된 배경음 결과가 일반적으로 0.11에서

0.14정도의 차이를 보인다는 사실에서 볼 때, 해당 차이는 상대적으로 매우 적은 차이라고 판단할 수 있다.

그림 2는 청취평가 결과를 도시하고 있다. 막대그래프는 각 실험군의 평균 점수를 의미하며, 평균 점수를 기준으로 위아래로 표시된 구간은 각 실험군의 표준편차를 나타낸다. 각 트랙별 막대그래프는 왼쪽부터 순서대로 원본 음원, 1-kHz 고역 통과 필터 처리 음원, 실시간 배경음 추출 결과 음원, 그리고 기준이 되는 일괄 배경음 추출 결과 음원을 나타낸다. 그림 2의 그래프를 살펴보면 모든 트랙에서 제안하는 실시간 배경음 추출 알고리즘이 기존의 일괄 배경음 추출 알고리즘과 유사한 결과를 보이는 것을 확인할 수 있으며, 특히 1, 3, 15번 트랙을 제외한 다른 모든 결과에서는 결과 데이터의 분포가 겹칠 정도로 유사함을 확인할 수 있다. 1, 3, 그리고 15번 트랙의 결과 또한 다른 비교 음원의 결과를 볼 때 충분히 기존 알고리즘과 유사한 특성을 보인다고 판단할 수 있다.

## V. 결론

본 논문에서는 단일 채널의 음향 신호에서 비음수 행렬 분해 기법을 통하여 배경음 성분을 추출하는 알고리즘에

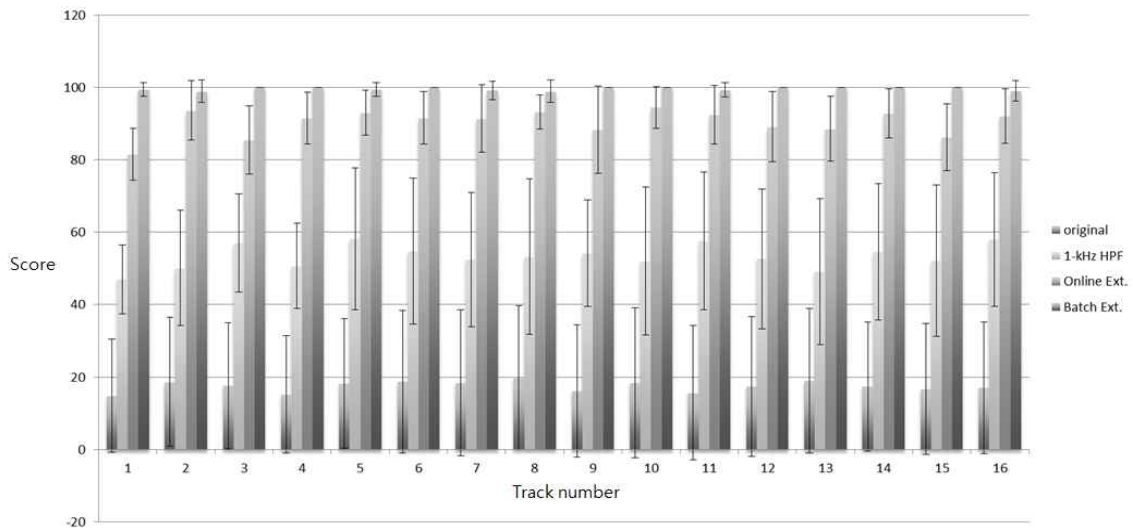


그림 2. 청취 평가 결과  
Fig. 2. Results of subjective listening tests

대해 살펴보았다. 또한, 이를 보완하여 일괄 처리가 아닌 실시간 처리를 통하여 배경음 성분을 추출할 수 있는 알고리즘을 제안하였다. 제안된 알고리즘은 실시간 비음수 행렬 분해 기법을 통하여 음향 신호를 분석하고, 분석되지 않는 나머지 신호를 배경음 성분으로 추출함으로써, 주파수 영역에 걸쳐 넓게 분포되어 있는 에너지를 가지는 성분을 추출해 낸다.

RWC 데이터베이스의 음악 신호를 이용하여 제안하는 알고리즘의 성능을 분석하였다. 본 알고리즘의 성능을 검증하기 위하여, 기존의 일괄처리기법을 통하여 처리된 신호의 성능 지표를 목표하는 결과로, 그리고 처리되지 않은 입력 신호의 성능 지표를 최저 기준값으로 하여 제안하는 알고리즘의 성능을 검증하였다. 이를 통하여 제안하는 알고리즘이 실시간 처리라는 제약 조건을 가지고 있음에도 불구하고, 기존의 일괄 처리 알고리즘의 결과와 유사한 결과를 보임을 확인할 수 있었다.

#### 참 고 문 헌 (References)

- [1] H. Chung, H. Shim, N. Hahn, and S. B. Chon, "Sound reproduction method by front loudspeaker array for home theater applications," *IEEE Transactions on Consumer Electronics*, vol. 50, no. 2, pp. 528-534, May 2012.
- [2] R. Irwan and R. M. Aarts, "Two-to-Five Channel Sound Processing," *Journal of Audio Engineering Society*, vol. 50, no. 11, pp. 914-926, 2002.
- [3] C. Avendano and J. M. Jot, "Ambience Extraction and Synthesis from Stereo Signals for Multichannel Audio Up-mix," *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Orlando, USA, 2002.
- [4] C. Uhle, A. Walther, O. Hellmuth, and J. Herre, "Ambience separation from mono recordings using non-negative matrix factorization," *Proc. AES 30th International Conference on Intelligent Audio Environments*, Mar. 2007.
- [5] D. D. Lee and H. S. Seung, "Learning the Parts of Objects by Non-negative Matrix Factorization," *Nature*, vol. 401, pp.788-791, Aug. 1999.
- [6] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," *Proc. Advances in Neural Information Processing 13 (NIPS' 2000)*, pp. 556-562, 2001.
- [7] A. Cichocki, R. Zdunek, A. H. Pahn, S. Amari, *Nonnegative matrix and tensor factorizations: applications to exploratory multiway data analysis and blind source separation*, John Wiley & Sons, Chichester, 2009.
- [8] S. Lee, S. H. Park, K.-M. Sung, "On-line nonnegative matrix factorization method using recursive least squares for acoustic signal processing systems," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Science*, vol. E94-A, no. 10, pp. 2022-2026, Oct. 2011.
- [9] S. Lee, "RLS-based on-line sparse nonnegative matrix factorization method for acoustic signal processing systems," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Science*, vol. E96-A, no. 5, pp. 980-985, May 2013.
- [10] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka, "RWC music database: popular, classical, and jazz music database," *Proceedings of the 3rd International Conference on Music Information Retrieval (ISMIR 2002)*, pp. 287-288, Oct. 2002.
- [11] G. Peeters, "A large set of audio features for sound description (similarity and description) in the CUIDADO project," *IRCAM*, 2004.

---

#### 저 자 소 개



이 석 진

- 2006년 : 서울대학교 전기컴퓨터공학부 공학사
- 2008년 : 서울대학교 전기컴퓨터공학부 공학석사
- 2012년 : 서울대학교 전기컴퓨터공학부 공학박사
- 2012년 ~ 2014년 : LG전자 CTO부문연구소 선임연구원
- 2014년 ~ 현재 : 경기대학교 전자공학과 조교수
- 주관심분야 : 음향신호처리, 배열신호처리, 적응신호처리, 음원분리