

청각 장애인을 위한 음악의 실시간 시각화 시스템 개발

최 두 일
 공주대학교 전기전자정보통신 공학부

Development of Real-Time Musical Visualizing System for the Deaf

Choi, Doo-Il
 Dept. of Electrical Electronic & Information Eng., Kongju Univ.

Abstract - We have developed a real time musical visualizing system works on PC with sound card and windows 95 or more. The system can extract some features of music signal and animate various screen -i.e. wave, spectrum, simulated water jet - according to extracted features. We have used the system at the concert for the deaf and contributed for deaf to understand the music.

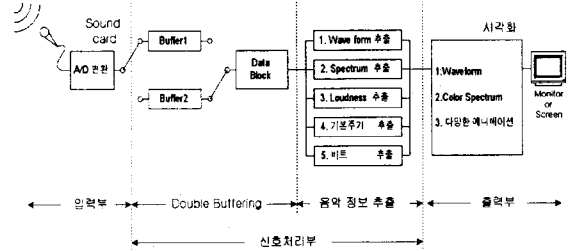


그림 1 실시간 음악시각화 시스템 전체 블록선도.

1. 서 론

청각 장애인을 위한 음악회가 공식적으로 개최된 적은 아직까지는 보고되지 않고 있었다. 1999년 9월 21일에 공주대학교에서 주최한 음악회는 주로 장애인을 위한 것이었으며 본 연구에서 개발된 시스템이 이용되어 청각 장애인의 음악에 대한 이해를 돕는데 기여하였다. [1] 일반적으로 청각계통에 장애가 있는 사람이 소리를 느끼기 위해서는 청각 보철 [2][3]을 시술받거나 청각 대신 촉각이나 시각을 사용한다. 음악회에서 현실적으로 수많은 청각 장애인에게 보철이나 촉각감각으로 음악을 느끼도록 하는 것은 불가능하므로 음악의 시각적 표현이 가장 현실적이라고 판단되었으며, 이를 목적으로 본 시스템이 개발되었다.

음악에는 파형과 스펙트럼 정보 외에도 다른 음향신호와 달리 강약, 고저, 박자 등의 정보가 포함되어 있으므로 이러한 정보를 추출하여 애니메이션 영상에 이용함으로써 음악에 대한 이해를 증진시킬 수 있다는 판단 하에 음악 정보 자동 추출 알고리즘을 개발하였고, 개발된 알고리즘을 적용하여 실시간 음악 시각화 시스템을 구현하였다.

2. 시스템 개발

본 연구에서 개발된 시스템의 전체 블록선도는 그림 1과 같다. 음악 데이터는 PC의 sound card를 이용하여 수집하였고, 실시간 처리를 위하여 double buffering 기법을 이용하였다. 수집된 음악 신호는 moving window를 취한 후 매 프레임마다 파형, 스펙트럼, loudness, 기본 주파수 및 박자 정보를 추출하였고 추출된 정보는 다양한 애니메이션에 이용되도록 하였다. 모든 과정은 GUI(graphic user interface) 환경 하에서 C++ 언어로 software적으로 구현하였다. 음악회 당시에 영상의 출력은 PC용 beam projector와 대형 스크린을 이용하였다.

다음은 각 단계별 개발 내용이다.

2.1 입력부

A/D 변환은 22.050KHz sampling rate, 16 bit/sample, mono로 수행되도록 하였다. 실시간 처리를 위한 double buffer size는 64에서 2048까지 선택 가능하도록 하였다. moving window는 hamming window를 사용하였고, window의 이동 간격은 1에서 window 크기까지 선택 가능하도록 설계하였다.

2.2 신호처리부

음악 신호에서 파형, loudness, spectrum 및 기본 주파수를 얻는 블록선도는 다음과 같다.

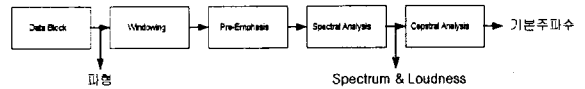


그림 2 음악 정보 추출 블록선도.

2.2.1 파형

음악 신호를 $s[n]$, 윈도우 함수를 $w[n]$, 윈도우 길이를 N , 윈도우의 이동거리를 M 이라 하면 k 번째 frame의 음악 파형은 식 (1)과 같다.

$$s_k[m] = \sum_{n=0}^N s[n]w[n - (kM + m)] \quad (1)$$

2.2.2 스펙트럼

k 번째 frame의 음악 스펙트럼은 식 (2)와 같다.

$$S_k[j\Omega] = F\{s_k[m]\} \quad (2)$$

여기서 $F\{\}$ 는 Fourier Transform

2.2.3 Loudness L

k 번째 frame의 음악 Loudness는 식 (3)과 같다.

$$L_k = \sum_{\Omega=0}^{2\pi} |S_k[j\Omega](dB)|^2 A[j\Omega] \quad (3)$$

여기서, $A[j\Omega]$ 는 인간 청각 계통의 주파수 특성이다.

2.2.4 기본 주파수 F_0

F_0 의 추출은 Autocorrelation방법, Cepstrum 분석법

등이 있다. Cepstrum 분석법을 이용한 F_0 의 추출은 다음과 같다.

$S_k[j\Omega] \rightarrow$ Power Spectrum $\rightarrow \log(\) \rightarrow$ 역 Fourier 변환 \rightarrow Cepstrum $\rightarrow F_0$

2.2.5 비트 B (박자의 속도)

음악의 빠르기를 자동으로 추출하기 위해서 다음의 가정을 기초로 한다.

가정 1 : 음악의 bit는 타악기(drum 등) 소리에 의해서 결정된다.

가정 2 : 타악기 소리의 주파수 대역은 일반 음악소리에 비해 상대적으로 고주파 영역에 에너지가 밀집되어 있으며, 백색 잡음과 유사한 성질을 보인다.

가정 3 : 타악기의 비트가 포함된 음악 구간은 다른 구간에 비하여 비정상(non-stationary)이며, 임펄스성 신호성분이 존재한다.

이상의 가정은 대부분의 음악에서 보여지는 일반적인 현상이며, 타악기가 없거나 특수한 타악기가 사용된 음악에 대해서는 적용되지 않는다.

이 가정을 적용한 실시간 비트 자동 추출 알고리즘은 표 1과 같다.

표 1 실시간 비트 자동 추출 알고리즘

```

step 1 : define and initialize variables
M : window size
fl : 타악기 저주파 차단 주파수
fh : 타악기 고주파 차단 주파수
E(n) : n번째 frame의 타악기 주파수 영역의 에너지
P(n) : n번째 frame의 peak값
Tframe : frame 시간(=window size/sampling rate)
Eaverage(n) : 이전 K frame의 평균 에너지
Paverage(n) : 이전 K frame의 평균 peak값
B : 타악기 bit 간격(전역값)
Blocal : 타악기 bit 간격(지역값)

step 2 : Present n-th frame of Music Signal
sn(0), sn(1), ..... , sn(M-1)

step 3 : E(n), Eaverage(n), P(n) 및 Paverage(n)를 계산
Eaverage(n) = 1/K ∑k=0K-1 E[n-k]
Paverage(n) = 1/K ∑k=0K-1 P[n-k]
E(n) = 1/(fh-fl) ∑f=flfh Sn2(f)
P(n) = max(sn(0), sn(1), ..... , sn(M-1))

step 4 : bit의 유무 결정 및 local bit interval 계산
if E(n) > α Eaverage(n) and P(n) > β Paverage(n)
    Blocal = Counter × Tframe
    Counter = 0
    B = B + gain (Blocal - B)
else
    Counter = Counter + 1
where α, β > 1 and gain << 1

step 5 : replace n by n+1 and goto step 2
    
```

2.2 출력부

파형과 스펙트럼은 기본적으로 출력하며, 신호처리부에서 추출된 음악 정보(L, F₀, B등)를 이용하여 분수, 우주여행, 폭죽놀이와 같은 애니메이션을 동작시킴으로써 마치 애니메이션 영상이 음악에 따라 춤을 추는 듯한 효과를 얻도록 하였다. 예로써 개발한 음악 분수 시뮬레이션에서는 L 로 분수의 수압(높이)을, F₀ 로 분수의 색깔을 그리고 B 로 분수의 좌우 흔들림 표현하도록 하였다. 이 외에 우주여행 애니메이션 영상을 개발하였고, 폭죽놀이 시뮬레이션이 개발 중에 있다. 분수의 시뮬레이션을 위한 알고리즘을 살펴보면 표 2와 같다.

이전에서는 L 로 분수의 수압(높이)을, F₀ 로 분수의 색깔을 그리고 B 로 분수의 좌우 흔들림 표현하도록 하였다. 이 외에 우주여행 애니메이션 영상을 개발하였고, 폭죽놀이 시뮬레이션이 개발 중에 있다. 분수의 시뮬레이션을 위한 알고리즘을 살펴보면 표 2와 같다.

표 2 음악 분수 애니메이션 알고리즘

```

step 1 : define variables
cx : 화면의 넓이
cy : 화면의 높이
N : 물방울 개수
Δt : 분수 동작 속도
G : 중력 가속도
V0(i) : i 번째 물방울의 초기 분출 속도
θ0(i) : i 번째 물방울의 초기 분출 각도
t(i) : i 번째 물방울의 분출후 경과 시간
θavg : 물방울의 평균 분출 각도
L : Loudness of Music
F0 : fundamental frequency of Music
B : Bit Time of Music

step 2 : 분수의 초기화
for each i (0<i<N)
Set V0(i) has uniform distribution in [ 0, Vmax ]
Set θ0(i) has gaussian distribution with zero mean
Set t(i) = 0
θavg = 0

step 3 : 물방울의 색 지정 및 평균 분출 각도 계산
j = j + 0.01/B
θavg = π/4 cos(2πj)
F0 로 분수의 색깔 표현 :
F0가 클수록 차가운 계열의 색으로 지정

step 4 : 다음 프레임의 각 물방울 위치 계산 및 그리기
for each i (0<i<N)
t(i) = t(i) + Δt
x = V0[i] cos(θ0[i]) Δt ;
y = V0[i] - 1/2 G Δt2 ;
if( 좌표(x, y)가 화면을 벗어났으면 ) {
Set V0(i) has gaussian distribution with L mean
Set θ0(i) has gaussian distribution with θavg mean
t(i) = 0
}
else {
좌표(x, y)에 지정된 색으로 각 물방울 그리기
}

step 5 : Find L, F0, B of next frame and goto step 3
    
```

3. 개발 결과

그림 3은 개발된 시스템의 출력 화면을 보이고 있다. 출력 화면의 상단에는 파형을 디스플레이하고 하단에는 스펙트럼 정보를 디스플레이 하도록 하였다. 스펙트럼 화면의 배경화면으로 음악 분수, 우주 여행 시뮬레이션 등을 선택하도록 설계되었다.

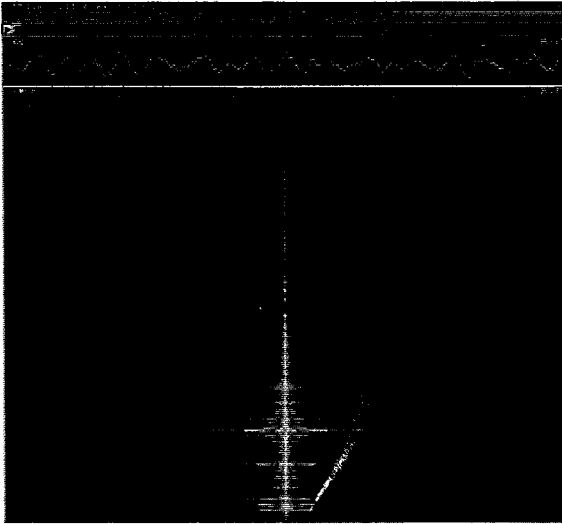


그림 3. 개발된 시스템의 출력 화면

4. 결 론

본 연구에서는 청각 장애인을 위한 음악회에서 장애인들의 음악 감상을 보조하기 위한 실시간 음악 시각화 시스템을 개발하였다. 개발된 시스템은 음악회에서 시연되었고 청각 장애인의 소감을 설문 조사한 결과 매우 만족했음을 알 수 있었다. 장애인들의 음악 감상시 감동의 극대화를 이루기 위해서는 향후에도 많은 아이디어가 도출되어야 할 것으로 생각된다.

[참 고 문 헌]

- [1] 중앙일보, "청각장애인에 소리 보여줘", 1999년 9월 28일자
- [2] Doo-Il Choi, Dong-Hyeok Kim, Sang-Hui Park, Seung-Hwa Beack, Dong-Min Woo, "Design of Speech Signal Processor for the Multi-Channel Auditory Prosthesis using Auditory Model", World Congress on Medical Physics and Biomedical Engineering, Vol. 29, pp.146, July, 1991, Kyoto, Japan.
- [3] 채대곤, 변정근, 최두일, 백승화, 박상희, "청각 장애인을 위한 음성신호의 자극패턴 추출에 관한 연구", 대한의용생체공학회 추계학술대회 논문집 Vol. 15, No. 2, pp. 175-179, 1993. 11