



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

석사학위논문

Max/MSP와 New Media Art를 이용한  
인터랙티브 멀티미디어 퍼포먼스 제작 연구  
(멀티미디어음악 작품 <Eternity>를 중심으로)

지도교수 김 준

동국대학교 영상대학원

멀티미디어학과 컴퓨터음악전공

임 준 형

2020

석사학위논문

Max/MSP와 New Media Art를 이용한  
인터랙티브 멀티미디어 퍼포먼스 제작 연구  
(멀티미디어음악 작품 <Eternity>를 중심으로)

임준형

지도교수 김준

이 논문을 석사학위 논문으로 제출함

2019년 12월

임준형의 음악석사(컴퓨터음악)학위 논문을 인준함

2020년 1월

위원장 박상훈



위원 정진현



위원 김준



동국대학교 영상대학원

## 목 차

I. 서 론 .....	1
1. 연구 배경 및 목적 .....	1
2. 사례 연구 .....	3
II. 기술 연구 .....	6
1. 사운드 시스템 연구 .....	6
1) 사운드 시스템 .....	6
2) tape music을 위한 사운드 디자인 .....	8
① additive synthesis .....	9
② frequency modulation synthesis .....	11
③ comb filter 음향효과 .....	13
④ sound source recreation .....	14
3) 실시간 사운드 프로세싱을 위한 음향효과 연구 과정 .....	17
① stereo delay 음향효과 .....	17
② flanger 음향효과 .....	19
③ chorus 음향효과 .....	20
④ granular synthesis 음향효과 .....	21
⑤ pitch shift 음향효과 .....	22
⑥ FFT 분석을 이용한 실시간 phase vocoder 음향효과 .....	24
가) FFT 분석을 이용한 pitch detection .....	24
나) pfft~오브젝트를 이용한 실시간 phase vocoder 패치 연구 .....	25
⑦ FFT 분석을 이용한 pfft~ngate~오브젝트 활용 .....	27

2. 영상 시스템 연구 .....	28
1) 영상 시스템 .....	28
2) After Effect를 이용한 영상 제작 .....	29
3) Max를 이용한 영상 제작 .....	31
4) Arena를 이용한 영상 시스템 .....	33
5) Syphon을 이용한 영상 데이터 전송 .....	34
3. 동작인식 시스템 연구 .....	35
1) 모션 트래킹 데이터 추출 .....	35
2) 모션 트래킹 데이터 매핑 .....	36
4. 인터페이스 PRIZONE 제작 연구 .....	41
1) Arduino를 이용한 인터페이스 시스템 구축 .....	41
2) PRIZONE의 Interactive Visualization을 위한 LED .....	46
5. 실시간 공연 시스템 연구 .....	47
1) 공연 시스템 .....	47
2) 실시간 인터랙션 시스템 구축 .....	48
① 사운드 인터랙션 시스템 연구 .....	48
② 영상 인터랙션 시스템 연구 .....	50
③ 인터페이스 인터랙션 시스템 연구 .....	51
Ⅲ. 연구 기술의 작품 적용 .....	52
1. 작품 소개 .....	52
2. 무대 및 영상 구성 .....	54
1) 무대 구성 .....	54
2) 영상 구성 .....	55

3. 연구 기술 적용 .....	57
1) A파트 .....	57
2) A' 파트 .....	58
3) B파트 .....	61
4) C파트 .....	63
5) D파트 .....	65
6) A" 파트 .....	67
4. 기술 적용의 예술적 효과 .....	68
IV. 결 론 .....	70
참 고 문 헌 .....	72
ABSTRACT .....	75
부록-1 : Rimbaud <L'Éternité> .....	77
부록-2 : 첨부 DVD 설명 .....	78

## 표 목 차

<표-1> 오브젝트를 통해 생성되는 파형의 종류 .....	10
<표-2> munger~오브젝트 각 파라미터 설명 .....	21
<표-3> grain pitch 파라미터 값 정리 .....	22
<표-4> Arena 영상효과 .....	33
<표-5> 동작에 따른 트래킹 안정성 실험 결과 .....	40
<표-6> 머리와 몸통의 최대, 최소 위치 값 .....	48
<표-7> 머리와 몸통의 최대, 최소에 따른 방향 .....	48
<표-8> 작품 <Eternity> 전체 구조 .....	53
<표-9> 영상 구성 .....	55
<표-10> A파트 구성 .....	57
<표-11> A' 파트 구성 .....	58
<표-12> B파트 구성 .....	61
<표-10> C파트 구성 .....	63
<표-10> D파트 구성 .....	65
<표-10> A" 파트 구성 .....	67

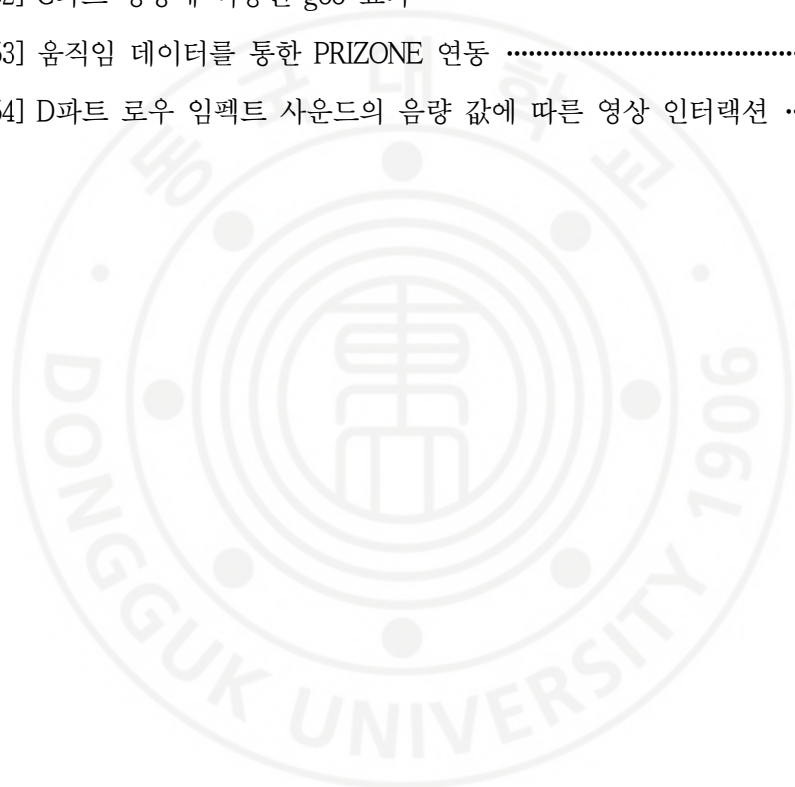
## 그 립 목 차

[그림-1] 뮤지컬 <빈센트 반 고흐> .....	3
[그림-2] 그룹 PROFORMA의 Kinect 퍼포먼스 .....	4
[그림-3] 사운드 시스템 설계도 .....	7
[그림-4] additive synthesis 예시 .....	9
[그림-5] additive synthesis 패치 .....	11
[그림-6] simple FM synthesis 패치 .....	12
[그림-7] comb filtering 음향효과 .....	13
[그림-8] comb filter 패치 구현 .....	14
[그림-9] groove~오브젝트와 buffer~오브젝트 패치 .....	15
[그림-10] groove~오브젝트를 적용한 파형 .....	16
[그림-11] stereo delay 패치 .....	17
[그림-12] flanger 패치 .....	19
[그림-13] chorus 패치 .....	20
[그림-14] granular synthesis 패치 .....	21
[그림-15] pitch shift 패치 .....	23
[그림-16] 푸리에 변환 .....	24
[그림-17] 실시간 phase vocoder 패치 .....	25
[그림-18] noise gate 패치 .....	27
[그림-19] 영상 시스템 .....	28
[그림-20] 음원의 주파수 스펙트럼 .....	29
[그림-21] 영상 인터랙션 구현 .....	30
[그림-22] sonogram 패치 .....	31



[그림-23] sonogram 영상 .....	32
[그림-24] Syphon 전송 패치 .....	34
[그림-25] Arena 최종 출력 .....	34
[그림-26] Kinect 구성도 .....	35
[그림-27] skeleton tracking 이미지 .....	36
[그림-28] NI-mate에서의 영역 설정 .....	37
[그림-29] 피타고라스 공식 .....	38
[그림-30] NI-mate에서의 format 설정 .....	38
[그림-31] NI-mate에서의 IP와 port 설정 .....	39
[그림-32] OSC 데이터 값 추출 패치 .....	40
[그림-33] PRIZONE 인터페이스 연동 시스템 .....	42
[그림-34] serial 통신 구현 패치 .....	42
[그림-35] Arduino IDE를 활용한 LED 제어 코딩 .....	43
[그림-36] duty cycle에 따른 PWM .....	44
[그림-37] 브레드보드를 이용한 Arduino와 LED 연결 도면 .....	45
[그림-38] PRIZONE 하드웨어 제작과정 .....	46
[그림-39] 공연 시스템 설계도 .....	47
[그림-40] udpreceive오브젝트를 이용한 사운드 인터랙션 패치 .....	49
[그림-41] Arena와 Max의 통신을 위한 OSC 주소 설정 .....	50
[그림-42] PRIZONE 인터랙션 패치 .....	51
[그림-43] 무대 시스템 구성 .....	54
[그림-44] A파트 사운드 프로세싱 패치 .....	57
[그림-45] A파트 목소리에 따른 영상 인터랙션 .....	58
[그림-46] tape music 소스 제작 패치 .....	59

[그림-47] A' 파트 영상효과에 따른 변화 과정 .....	60
[그림-48] B파트 phase vocoder + flanger 패치 .....	61
[그림-49] B파트 사운드 소스 편집 과정 .....	62
[그림-50] B파트 사운드 소스 연결 과정 .....	62
[그림-51] C파트 사운드 및 인터랙션 패치 .....	64
[그림-52] C파트 영상에 적용된 goo 효과 .....	64
[그림-53] 움직임 데이터를 통한 PRIZONE 연동 .....	65
[그림-54] D파트 로우 임팩트 사운드의 음량 값에 따른 영상 인터랙션 .....	66



# I. 서론

## 1. 연구 배경 및 목적

현대 과학기술의 발달은 사회구조의 변화를 이끌고 온 과학기술 혁명<sup>1)</sup>에서부터, 인간이 삶을 향유하는 방식을 변화시키기까지 우리 사회의 모습을 다양하게 변화시키고 있다. 예술의 영역 또한 과학 기술의 발달에 따른 매체의 변화가 이루어지고 있다. 컴퓨터아트(computer art)<sup>2)</sup>, 인터랙티브아트(interactive art)<sup>3)</sup>, 미디어아트(media art)<sup>4)</sup> 등의 현대 예술 명칭들을 통합한 용어인 ‘뉴미디어아트’는 컴퓨터를 기반으로 한 새로운 미디어의 특징을 지닌 예술 장르를 부르는 명칭이다. 디지털 기술의 발달로 인해 3D 프로젝션 매핑<sup>5)</sup>, 홀로그램<sup>6)</sup>, LED(Light Emitting Diode)<sup>7)</sup> 스크린 등 새로운 기술을 이용한 예술을 주위에서 접할 수 있다. 공연예술(performing art)에서도 이러한 매체의 변화가 이루어지고 있다. 무대 위에서 여러 뉴미디어 기술의 사용은 공연예술의 특징이었던 시간과 공간적인 제약을 벗어나도록 한다. 이러한 새로운 매체의 출현은 이전 예술 작업이 지녔던 방법론적인 측면들을 확장 및 재조합하여 다양한 형태의 예술을 창조해내고 있다.

본 논문에서 설명하는 작품 <Eternity>는 뉴미디어와 연극을 융합한 실시간 인터랙티브 멀티미디어 퍼포먼스 작품이다. 배우의 음성을 실시간 사운드

- 
- 1) 과학지식과 기술체계상에서 획기적인 변화를 의미한다.
  - 2) 컴퓨터의 기능을 음향, 그래픽, 영상작품 등에 이용하는 예술을 말한다.
  - 3) 관객과의 상호작용을 유도하는 예술을 말한다.
  - 4) 사진, 전화, 영화 등의 발명 이후 신기술을 활용하는 예술을 말한다.
  - 5) 대상물의 표면에 빛으로 이루어진 영상을 투사하여 변화를 줌으로써, 현실에 존재하는 대상이 다른 성격을 가진 것처럼 보이도록 하는 기술이다.
  - 6) 홀로그램을 통해 사물을 평면상에 입체로 기록한 매체를 일컫는다.
  - 7) 발광 다이오드, 순방향으로 전압을 가했을 때 발광하는 반도체 소자이다.

프로세싱(real sound processing)<sup>8)</sup>하여 영상과 LED에 연동되도록 시스템을 설계하였다. 이를 통하여 연출적인 표현의 범위를 확장시켜 관객들에게 새로운 경험을 주는 데에 목적을 둔다. 또한 뉴미디어를 사용한 인터페이스 및 영상을 제작하고, 연기(acting)에 따른 매체의 인터랙션을 통해 연출적인 다양성에 대한 연구를 진행하였다.

본 연구는 연극, 음악, 영상, LED의 특성을 관찰하고 활용하여 연극적인 제약<sup>9)</sup>을 뉴미디어 적인 요소로 해결하고 확장하는 예술 작품을 제작하기 위하여 시작하였다.

---

8) 사운드 시그널을 실시간으로 변조시켜주는 과정을 의미한다.

9) 연극은 실시간으로 공연되기 때문에 시간의 제약을 받는다. 연극의 공간 또한 무대 위 관객들에게 보이는 공간만이 배우의 행동범위가 되기에 제약이 크다.

## 2. 사례 연구

공연예술에서 뉴미디어의 활용은 다양하게 시도되고 있다. 보편적으로 공연에서의 미디어는 단순히 배경을 설정하는 목적으로 활용되었다. 그러나 현대 공연에서는 공간과 오브제(object)<sup>10)</sup>가 지니고 있는 연출적 한계를 미디어와 결합하여 극복해 준다. 인터랙티브 미디어(interactive media)<sup>11)</sup>를 활용하여 실시간으로 획득한 데이터를 영상과 결합하여 표현을 확장해 나가는 등, 공연 내에서 미디어는 연출에 있어서 다양한 역할을 해나가고 있다. [그림-1]은 뮤지컬 <빈센트 반 고흐>의 한 장면이다.



[그림-1] 뮤지컬 <빈센트 반 고흐>

10) 본래의 용도에서 분리하여 작품에 사용함으로써 새로운 느낌을 일으키는 상징적 기능의 물체를 이르는 말이다.

11) 매체간의 상호작용을 적용한 미디어라 한다.

이 작품은 프로젝션 매핑 기술을 사용한 미디어 파사드(media facade)<sup>12)</sup> 뮤지컬 작품이다. 벽과 오브제에 투사된 영상의 전환을 통하여 극의 진행을 도와주었으며, 배우의 움직임에 따라 연동된 인터랙티브 미디어가 활용되었다. 그러나 장면의 전환에 따라 약속된 동작을 하는 인터랙션이었기에 자연스러운 인터랙션이라고 보기 어렵다. 이러한 점을 보완하기 위해 센서를 이용한 실시간 인터랙티브 미디어에 대한 시도들이 이어지고 있다.



[그림-2] 그룹 PROFORMA의 Kinect 퍼포먼스

[그림-2]는 PROFORMA 그룹의 실시간 Kinect<sup>13)</sup> 퍼포먼스이다. 무용수가 각기 다른 오브제를 사용하여 춤을 추는 동안 Kinect를 이용하여 신체 데이터를 수집한 뒤, 음악에 맞춰 기하학적인 모양으로 시각화하였다. 실시간으로 변화하는 형체를 통해 무용수의 움직임을 확장시켜 극대화 시켜준다.

---

12) 건물 외벽을 활용하여 미디어 기능을 구현하는 것을 말한다.

13) Microsoft에서 개발한 Xbox의 주변기기이며, 이용자가 컨트롤러 없이 신체를 활용하여 게임과 엔터테인먼트를 경험 할 수 있다.

사례연구를 통해 볼 수 있듯이, 뉴미디어는 공연예술의 다방면에 걸쳐 활용되고 있다. 이는 배우, 배경, 주제, 의도에 대한 연출을 확장시키며 고전적인 작품과는 다른 극적인 효과를 줄 수 있다.

본 연구의 뉴미디어 기술은 연극 공연에 있어 배우의 대사와 움직임에 대한 정보를 활용하였으며, 배우의 연기에 대한 자연스러운 인터랙션 기술의 적용을 목표로 작품을 제작하였다. Kinect와 마이크를 통해 배우와 실시간 인터랙티브가 이루어진 미디어로 연출을 확장시키며, 기존의 연극적인 제약을 멀티미디어적인 요소로 해결하는, 뉴미디어 연극을 지향하는 멀티미디어 퍼포먼스의 제작을 시도하고 있다.

## II. 기술 연구

### 1. 사운드 시스템 연구

#### 1) 사운드 시스템

작품 <Eternity>는 tape music<sup>14)</sup>과 배우의 대사를 실시간 프로세싱한 사운드를 사용하여 제작하였다. tape music은 additive synthesis, frequency modulation synthesis, comb filter, groove~오브젝트, sound source recreation을 사용하여 제작하였고, 목소리의 실시간 사운드 프로세싱에는 stereo delay, flanger, chorus, granular synthesis, pitch shift, 실시간 phase vocoder, pfft~ngate~오브젝트가 사용되었다. 위의 음향효과들은 모두 Max/MSP(이하 Max)<sup>15)</sup>를 통해 구현 및 제작하였으며, 제작된 사운드는 Logic pro X<sup>16)</sup>를 통하여 편집해주었다.

[그림-3]은 tape music과 대사의 실시간 사운드 프로세싱을 위해 제작한 사운드 시스템 설계도이다. 배우의 대사는 무선 핀 마이크와 송수신기<sup>17)</sup>를 통해 오디오 인터페이스(audio interface)<sup>18)</sup>로 입력되어 컴퓨터로 전송되었다. 전송된 사운드는 Max로 전달되어 곡의 진행에 따라 다른 음향효과를 주었다. tape music과 실시간 프로세싱된 사운드는 음향 콘솔을 거쳐 스피커로 출력된다.

14) 전자음악 초기의 한 분류로 구체 음악, 구상 음악을 말한다. 본 논문에서는 합성한 사운드 소스를 편집하여 제작한 음악을 의미한다.

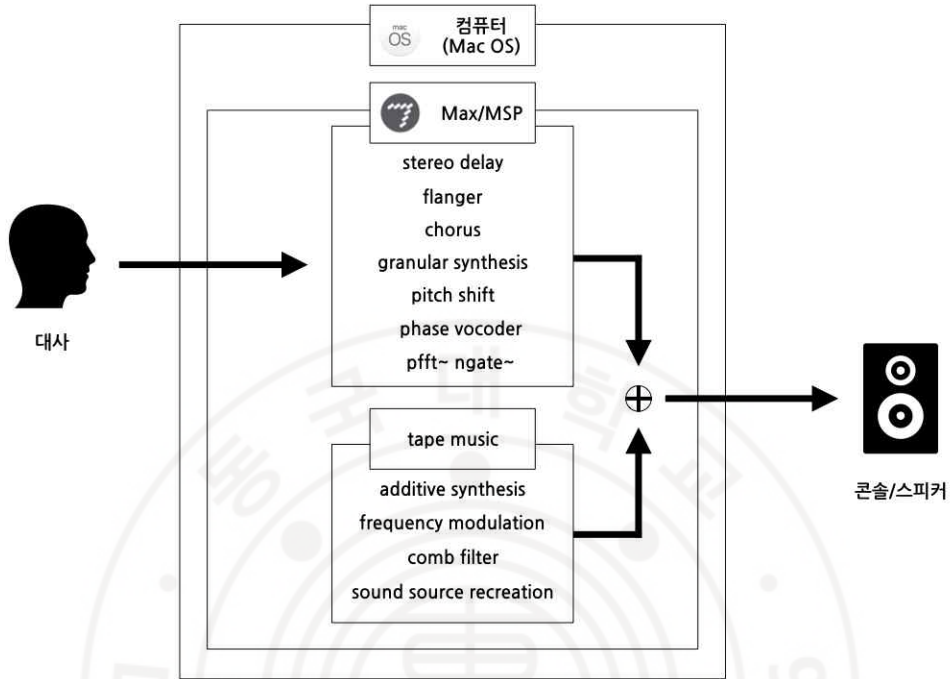
15) Cycling74에서 개발한 인터랙티브 미디어를 구현하는 프로그램이다. 데이터의 연산처리 및 프로그래밍이 가능한 Max와 음향 시그널 데이터 처리가 가능한 MSP, 그리고 real-time video 및 2D/3D 그래픽을 다루는 Jitter로 나누어져 있다.

16) mac OS에서 사용가능한 DAW(Digital Audio Workstation)

17) 사운드를 전송하고, 전송받는 기기

18) 아날로그 신호를 디지털 신호로 변환하며, 외부로 출력하는 녹음용 장비이다. 다양한 입출력 단자를 가지고 있어 음악 작곡, 악기 녹음 등 활용의 폭이 넓다.





[그림-3] 사운드 시스템 설계도

## 2) tape music을 위한 사운드 디자인

tape music으로 제작된 사운드는 크게 두 가지 방식으로 나뉜다.

첫째는 Max를 통해 사운드를 합성하여 제작하였다. 기본적인 사운드는 additive synthesis, multiple modulation<sup>19)</sup>, noise<sup>20)</sup>에 comb filter와 frequency feedback modulation을 거친 프로세싱 등으로 제작하였으며, amp envelope<sup>21)</sup>와 panning<sup>22)</sup>을 조정하여 현장감 있는 사운드를 구현하였다.

둘째는 사운드 프로세싱을 거친 소스를 편집하여 제작하였다. 오리지널 소스에 groove~오브젝트를 사용하여 reverse<sup>23)</sup> 해주거나 재생속도를 바꾸어 주었다. 또한 기존의 사운드 파형을 잘게 쪼개 이어 붙이거나, 재생속도를 변조시켜 사운드를 제작하였다.

---

19) 여러 가지 변조가 병합된 기술

20) Max에서 의도적으로 제작한 소음

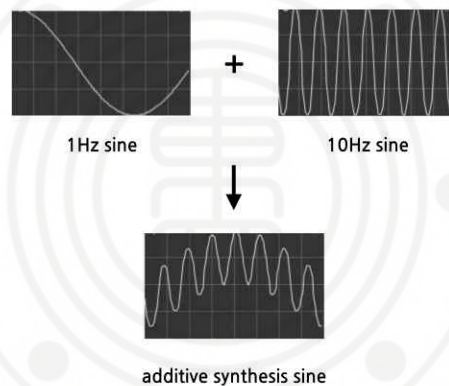
21) 시간에 따른 음량 값의 변화

22) 스테레오 시스템 안에서 청취자에게 들리는 소리의 위치를 정해주는 용어이다.

23) 파형의 시작과 끝 지점을 뒤집어준다.

### ① additive synthesis

가산 합성법은 여러 개의 개별 oscillator<sup>24)</sup>를 활용하여 배음(partial)<sup>25)</sup>을 생성하고, 배음 간의 위상<sup>26)</sup> 변화를 이용하여 새로운 소리를 만들어내는 합성법을 의미한다. Max에서 +~오브젝트<sup>27)</sup> 혹은 \*~오브젝트<sup>28)</sup>를 이용하여 구현할 수 있다. [그림-4]는 cycle~오브젝트를 이용하여 생성된 1Hz의 sine 파형과 10Hz의 sine 파형을 각각 보여주고 가산 합성법을 통하여 합성된 파형이다.



[그림-4] additive synthesis 예시

24) 사운드 신호를 생성하는 발진기

25) 하나의 음을 구성하는 여러 부분음들 중, 기본음보다 높은 정수배의 진동수를 갖는 음들의 모음

26) 반복되는 파형의 한 주기에서 첫 시작점의 각도 혹은 어느 한 순간의 위치를 말한다.

27) Max에서 사운드 시그널을 더해주는 오브젝트

28) Max에서 사운드 시그널을 곱해주는 오브젝트

Max에서는 cycle~오브젝트 이외에도 여러 다른 종류의 파형을 불러올 수 있는데, <표-1>은 오브젝트에 따라 생성할 수 있는 oscillator 파형을 정리하였다.

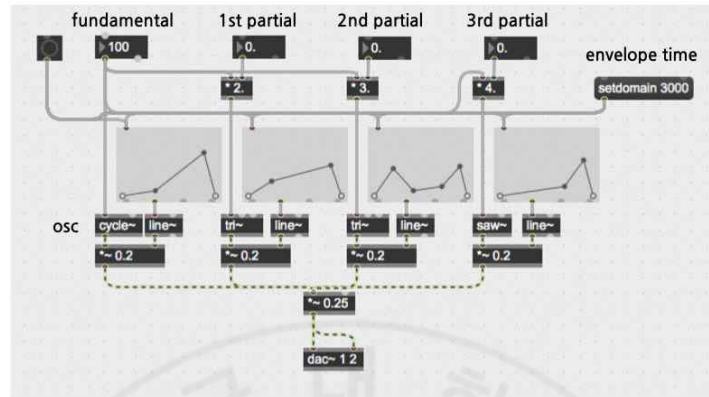
<표-1> 오브젝트를 통해 생성되는 파형의 종류

cycle~	사인파
cos~	코사인파
saw~	톱니파
rect~	사각파
tri~	삼각파

파형들의 주파수 값과 amp envelope를 설정한 뒤, additive synthesis를 통해 사운드를 합성하였다. 새로운 배음 구조를 만들기 위하여 근음(fundamental frequency)<sup>29)</sup>의 주파수를 기준으로 정수배의 값을 각각의 oscillator의 주파수 값에 입력했다. [그림-5]는 사운드 제작에 사용된 additive synthesis를 구현한 패치이다.

---

29) 기준이 되는 음



[그림-5] additive synthesis 패치

## ② frequency modulation synthesis

frequency modulation(이하 FM) synthesis는 주파수를 변조하는 사운드 합성 방식이다. FM synthesis는 스탠포드 대학교의 John Chowning 교수가 개발하였다.<sup>30)</sup> FM synthesis를 구현하기 위해서는 carrier oscillator<sup>31)</sup>와 modulation oscillator<sup>32)</sup>가 필요하다. modulation oscillator의 오디오 시그널은 carrier oscillator의 주파수 값에 더해져 변조를 일으킨다. 두 개의 oscillator만으로도 많은 side band<sup>33)</sup>를 생성하여 독특한 음색을 만들어낼 수 있다. modulation oscillator의 주파수가 사람의 가청주파수<sup>34)</sup> 영역보다 낮아지면 LFO<sup>35)</sup>라 부르며, 비브라토와 같이 주파수 폭의 변화를 사람이 느낄 수 있다.

30) J.M. Chowning. "The Synthesis of Complex Audio Spectra by Means of Frequency Modulation" (J. Audio Eng. Soc. 1973) 참조

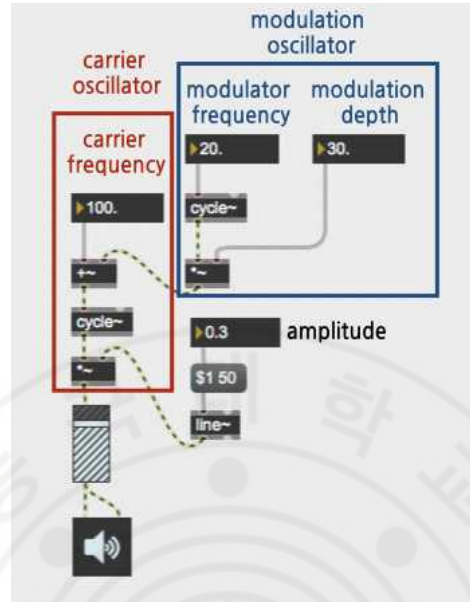
31) 기준 매개의 발진기

32) 적용하는 발진기

33) 기음을 기준으로 주변에 생기는 배음들

34) 사람의 귀가 소리로 느낄 수 있는 주파수 영역

35) Low Frequency Oscillator, 20Hz 이하의 주파수 대역을 지닌 발진기



[그림-6] simple FM synthesis 패치

[그림-6]은 기본적인 FM synthesis를 구현한 패치이다. FM synthesis는 여러 가지 알고리즘에 따라 연결하는 방식으로 나눌 수 있는데, 직렬식(serial)<sup>36)</sup>, 병렬식(parallel)<sup>37)</sup>, 피드백(feedback)<sup>38)</sup>의 방식이 존재한다. 본 작품에서는 FM 사운드 합성법을 다양한 알고리즘에 따라 조합하여 곡의 구성에 어울리는 다채로운 사운드를 구현하였다.

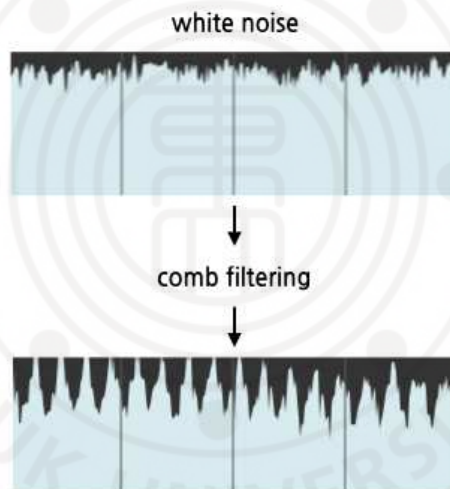
36) 한 줄로 배열되어있는 방식

37) 나란히 벌려서 배열되어있는 방식

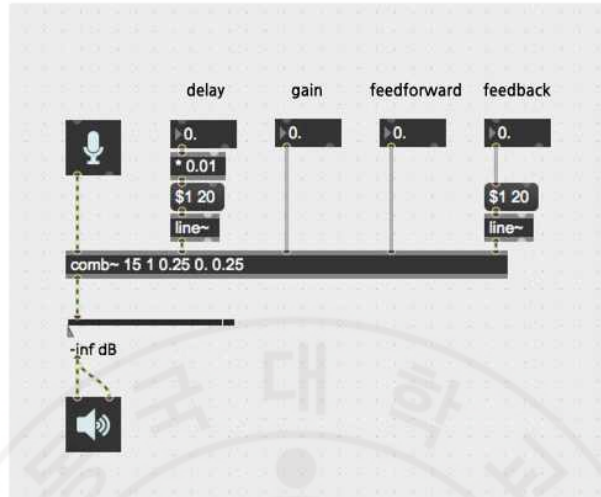
38) 일어난 결과가 다시 원인에 영향을 미치는 방식

### ③ comb filter 음향효과

comb filtering 효과는 직접음과 간섭음이 미세한 시간의 차이에 의해 겹쳐지는 효과이다. 주로 마이크 사이의 거리에 따라 발생하는 현상으로 음색의 왜곡이 일어난다. [그림-7]은 white noise에 comb filtering 효과를 입힌 파형의 모습이다. X축은 주파수이며 Y축은 음량 값을 의미한다. 그림과 같이 파형이 겹쳐지며 특정 주파수에서 상쇄와 보강이 일어난다. 머리빗(comb)의 모양과 유사하여 comb filter라고 불린다.



[그림-7] comb filtering 음향효과



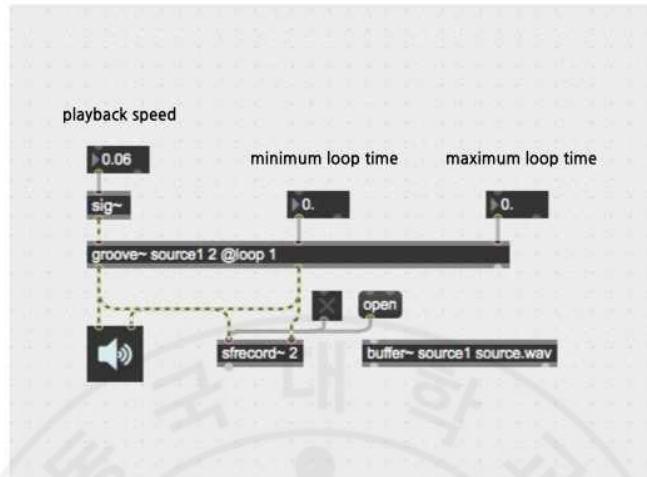
[그림-8] comb filter 패치 구현

위 그림은 comb filtering 효과를 Max로 구현한 패치이다. comb~오브젝트를 사용하며, delay 파라미터와 feedback 파라미터를 통해 음색과 음높이를 변화시킨다. 극장에서의 comb filtering 효과는 소리 왜곡으로 인하여 피드백이 생기는 현상이라고 보지만, 사운드 프로세싱에서는 차갑고 날카로운 금속성의 사운드를 표현하기 위해 의도적으로 사용한다.

#### ④ sound source recreation

합성된 사운드와 기존의 사운드 소스를 이용하여 새로운 사운드를 제작하였다. [그림-9]는 buffer~오브젝트와 groove~오브젝트를 이용하여 사운드의 재생속도를 변형해주는 패치를 Max로 구현하였다.

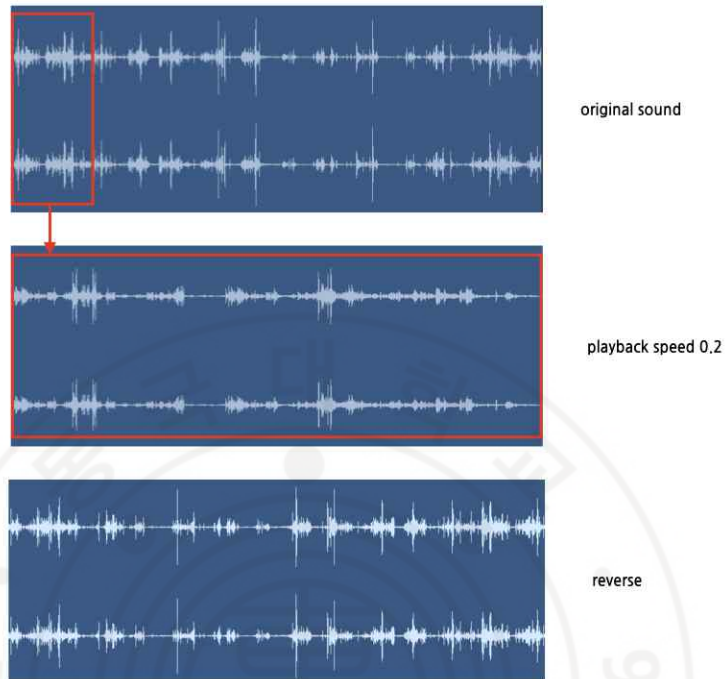




[그림-9] groove~오브젝트와 buffer~오브젝트 패치

buffer~오브젝트를 이용하여 사운드 샘플을 메모리에 저장한 후 재생한다. buffer~오브젝트를 통해 불러온 사운드 샘플은 groove~오브젝트의 playback speed를 조정해줌으로써 재생방향과 재생속도를 바꿔줄 수 있다. 이렇게 프로세싱된 사운드를 sfrecord~오브젝트를 사용하여 컴퓨터의 디스크에 wav audio file(WAV)<sup>39)</sup> 파일로 저장해주었다. [그림-10]은 오리지널 사운드와 groove~오브젝트를 통하여 프로세싱된 사운드 파형을 보여준다. 재생속도를 늦춰 기존 사운드보다 웅장하고 어두운 느낌을 표현하였다.

39) 윈도우가 기본으로 지원하는 비압축 웨이브폼 오디오 포맷이다.



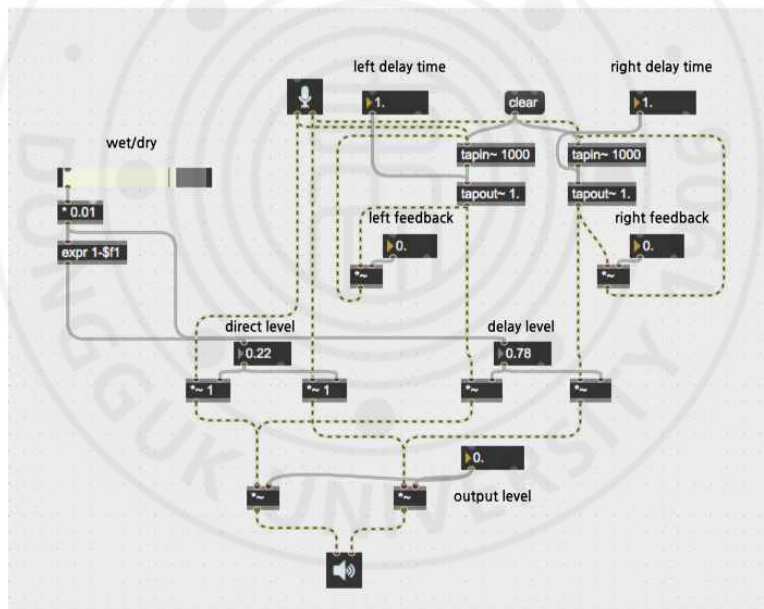
[그림-10] groove~오브젝트를 적용한 파형

groove~오브젝트와 플러그인(plugin)을 활용하여 변조된 사운드 소스를 Logic pro X를 통하여 짧게 자르거나 여러 사운드를 재배치함으로써 새로운 사운드를 연출하였다.

### 3) 실시간 사운드 프로세싱을 위한 음향효과 연구 과정

사람의 목소리는 다채로운 배음 구조를 지니고 있다. 배우는 장면에 대한 목표를 달성하기 위하여 넓은 스펙트럼의 정서를 표현하며 연기를 한다. 이러한 배우의 대사에 stereo delay, flanger, chorus, granular synthesis, pitch shift, phase vocoder 등의 실시간 사운드 프로세싱을 통해 사운드 디자인하였다.

#### ① stereo delay 음향효과



[그림-11] stereo delay 패치

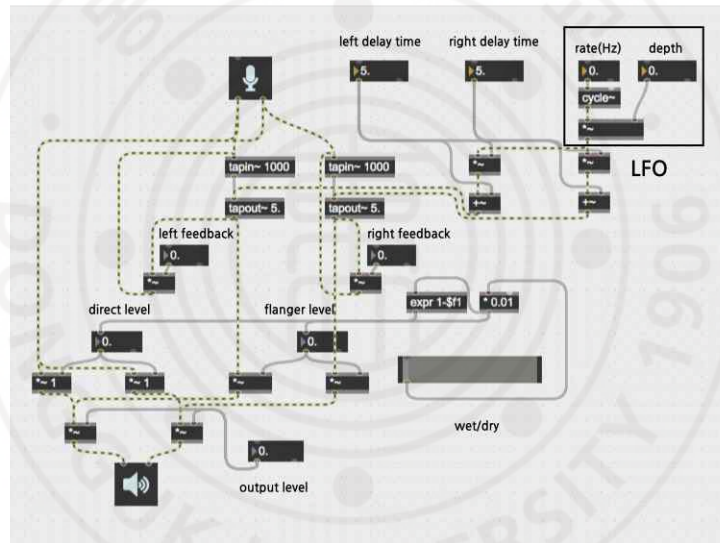
delay는 사운드 시그널을 지연하여 출력하는 사운드 음향효과이다. [그림-11]은 delay 음향효과를 Max로 구현한 패치이다. delay 음향효과는 tapin~오브젝트와 tapout~오브젝트를 이용하여 구현한다. tapin~오브젝트와 tapout~오브젝트의 시간 단위는 ms(밀리세컨드)<sup>40)</sup>이다. tapin~오브젝트는 사운드 시그널을 지연시키기 위하여 저장하는 공간이며 시간이다. tapout~오브젝트는 지연시간을 조절하는 역할을 한다. tapin~오브젝트의 공간이 가득 차게 되면 자동으로 사운드 시그널을 출력한 뒤 새로운 사운드 시그널을 입력받는다. 출력된 사운드 시그널은 tapout~오브젝트에서 지정한 시간만큼 지연하여 사운드를 내보낸다. tapout~오브젝트에서 나온 시그널을 tapin~오브젝트에 입력시키는 비율을 feedback amount라 부른다. expr<sup>41)</sup>오브젝트를 사용하여 오리지널 사운드와 프로세싱된 사운드의 비율을 조절할 수 있다. 원하는 디자인에 따라 왼쪽과 오른쪽의 지연 시간을 다르게 설정하여 사용할 수 있는데, 이를 stereo delay 혹은 ping pong delay라 부른다.

---

40) Max에서 계산식을 통하여 산수를 할 수 있는 오브젝트

## ② flanger 음향효과

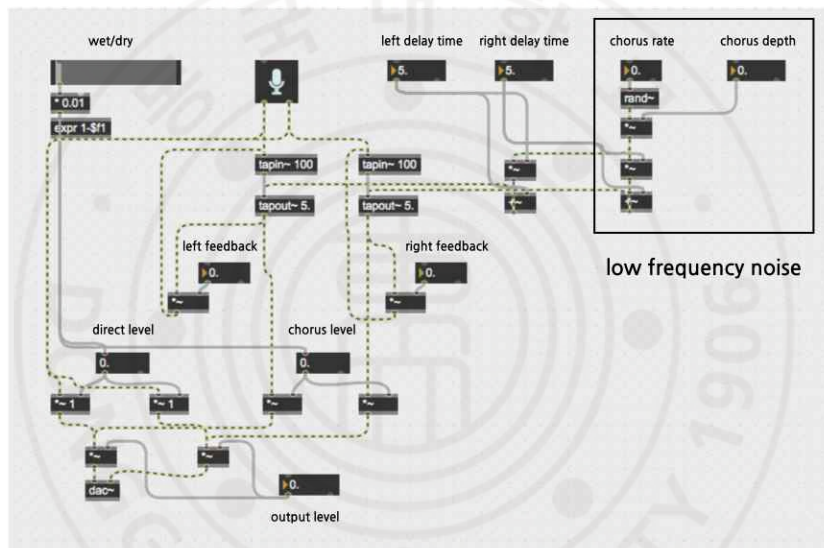
flanger는 delay 계열의 모듈레이션 음향효과이다. 두 가지의 동일한 사운드를 조금의 시간적 차이를 두고 출력시키면, 위상 차이에 의해 음색이 변하는 현상을 응용하여 사용하는 효과이다. 지연 시간을 실시간으로 변화하여 flanger 효과를 구현할 수 있는데, 기존 stereo delay 음향효과를 발전하여 delay time 파라미터(parameter)에 LFO 모듈레이션을 적용함으로써 구현한다.



[그림-12] flanger 패치

### ③ chorus 음향효과

chorus는 delay 계열의 모듈레이션 음향효과이며, 여러 사운드를 동시에 출력할 때 음원의 피치나 위상이 미세하게 다른 상태이다. chorus 음향효과는 delay time에 rand~오브젝트<sup>42)</sup>를 이용하여 생성한 noise를 모듈레이션에 적용하여 구현하였다.

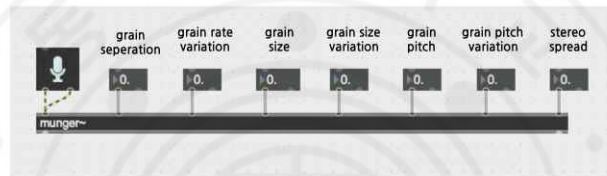


[그림-13] chorus 패치

42) 무작위의 사운드 시그널을 출력하는 오브젝트이다.

#### ④ granular synthesis 음향효과

granular synthesis는 입력된 사운드를 작은 시간 단위의 grain(샘플 조각)으로 쪼개어 재합성하는 소리 합성 방법이다. ms로 나뉜 여러 파형의 grain들이 새롭게 생성되며, 오리지널 사운드의 변형이 이루어진다. [그림-14]와 같이 Max에서 munger~오브젝트를 이용하여 구현할 수 있고 7개의 파라미터로 조절한다.



[그림-14] granular synthesis 패치

munger~오브젝트의 각 파라미터가 의미하는 바는 아래 <표-2>와 같다.

<표-2> munge~오브젝트 각 파라미터 설명

파라미터 명칭	파라미터 기능
grain separation	샘플 조각 간격
grain rate variation	샘플 조각 간격의 변화 값
grain size	샘플 조각 크기
grain size variation	샘플 조각 크기의 변화 값
grain pitch	샘플 조각의 음정
grain pitch variation	샘플 조각의 음정 변화 값
stereo spread	샘플 음들의 스테레오 이미지 정도 값

munger~오브젝트의 파라미터 중, grain pitch는 2의 제곱근으로 계산하여 음정 값을 변화시킨다.  $2^n$ 에서 지수의 값을 0으로 하는 원음을 1로 잡으면, 한 옥타브 위는 2, 두 옥타브 위는 4이며, 반대로 0.5는 한 옥타브 아래, 0.25는 두 옥타브 아래이다. [표-3]은 grain pitch 파라미터 값을 정리한 표이다.<sup>43)</sup>

[표-3] grain pitch 파라미터 값 정리

파라미터	$2^{-n}$	$2^{-2} = 0.25$	$2^{-1} = 0.5$	$2^0 = 1$	$2^1 = 2$	$2^2 = 4$	$2^n$
음역	n옥타브 아래	2옥타브 아래	1옥타브 아래	원음	1옥타브 위	2옥타브 아래	n옥타브 위

### ⑤ pitch shift 음향효과

pitch shift는 음색의 왜곡 없이 음높이를 변화시키는 음향효과이다. Max에서는 ZTX방식<sup>44)</sup>의 pitchshift~오브젝트를 사용하여 구현한다. quality를 통하여 pitch shift 음향효과의 품질을 지정할 수 있다. 음의 높낮이는 cent<sup>45)</sup> 단위로 조절할 수 있으며, ms 단위로 glide<sup>46)</sup> 효과를 구현할 수 있다. [그림-15]는 Max에서 pitchshift~오브젝트를 사용하여 pitch shift 음향효과를 구현한 패치이다.

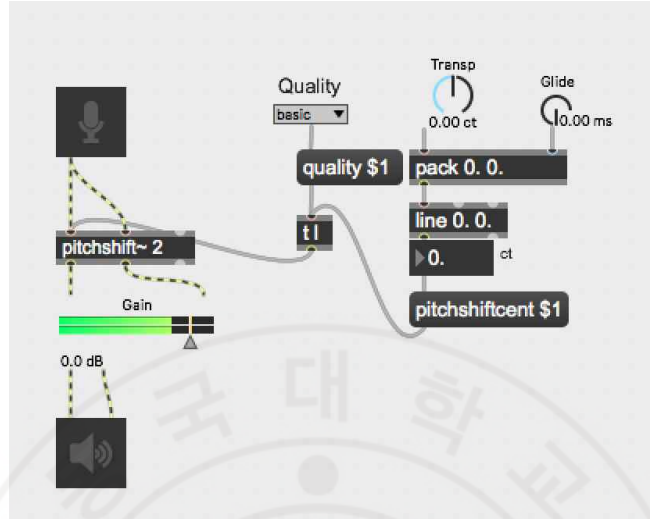
43) 한승욱 피아노의 실시간 프로세싱을 이용한 멀티미디어음악 제작 연구 멀티미디어음악 「작품 〈Midnight Peace〉를 중심으로」, 동국대학교 영상대학원: 멀티미디어학과, 석사학위논문, 2018, p. 4

44) ZTX는 고품질 음원 시간 스트레칭, 피치 변화, 피치 보정을 위한 디지털 신호처리 기술이다.

45) 음정의 단위이며, 1200cent는 1옥타브이다.

46) 한 음에서 다른 음으로 서서히 변화하는 효과이다.



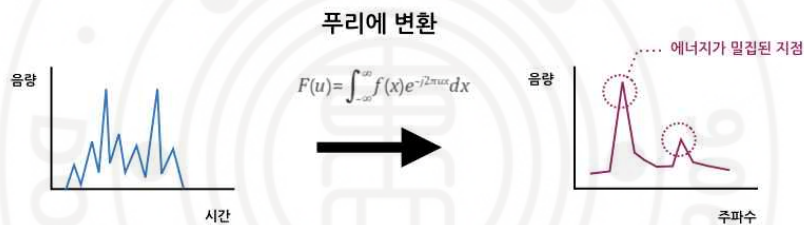


[그림-15] pitch shift 패치

## ⑥ FFT 분석을 이용한 실시간 phase vocoder 음향효과

### 가) FFT 분석을 이용한 pitch detection

음높이를 분석하여 검출하는 방법에는 시간(time)에 따른 분석과 주파수(frequency)에 따른 분석으로 나눌 수 있다. 시간에 따른 분석은 신호 파형을 스펙트럼(spectrum)<sup>47)</sup>화하지 않고 직접적으로 처리하는 방법이다. 주파수에 따른 분석은 입력된 사운드 시그널의 주파수 스펙트럼을 활용하여 분석하는데, 특히 에너지가 밀집된 지점에 중점을 두고 분석한다.<sup>48)</sup>



[그림-16] 푸리에 변환

[그림-16]에서 볼 수 있듯이, 푸리에 변환(Fourier transform)은 시간 영역의 신호를 주파수 영역으로 변환함으로써 주파수 분석의 기초가 된다. 작품에 사용된 고속 푸리에 변환인 FFT(Fast Fourier Transform)는 푸리에 변환 방식을 빠르게 연산한 것이다. FFT는 음정과 음정 간의 미세한 차이를 구분하는 데에 유용하다.<sup>49)</sup>

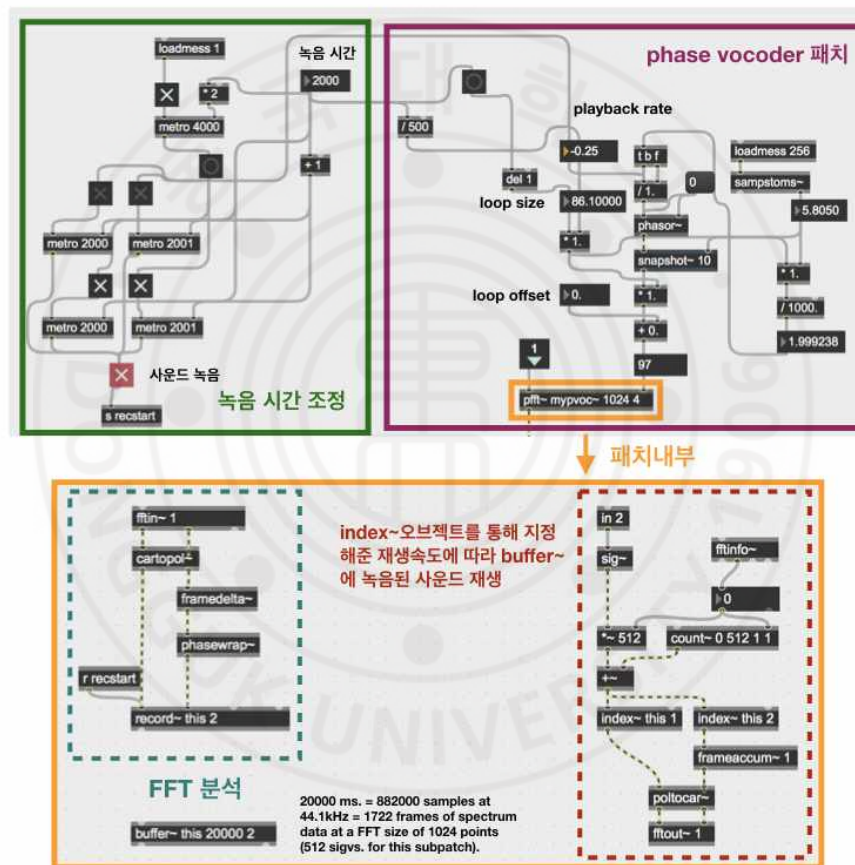
47) 특성을 한눈에 알아볼 수 있도록, X축과 Y축을 나눈 그래프이다.

48) 유효근, 「FFT변환, 음높이, LPC계수 및 LPC Cepstrum을 이용한 음정 분석」(건국대학교 산업대학원 전자공학과, 1997) p. 13

49) 유효근, 「FFT변환, 음높이, LPC계수 및 LPC Cepstrum을 이용한 음정 분석」(건국대학교 산업대학원 전자공학과, 1997) p. 38

나) pfft~오브젝트를 이용한 실시간 phase vocoder 패치 연구

[그림-17]은 pfft~오브젝트를 이용하여 실시간 phase vocoder를 구현한 패치와 pfft~ mypvoc~오브젝트의 패치 내부이다.



[그림-17] 실시간 phase vocoder 패치

record~오브젝트를 사용하여 녹음된 사운드는 buffer~오브젝트에 저장되어 패치 안에서 FFT 분석이 이루어진다. FFT 분석을 거쳐 저장된 데이터는 프레임(frame)<sup>50)</sup> 단위로 읽어 재생한다. loop size단위 또한 프레임 단위를 사용한다. 44100 sample rate<sup>51)</sup>를 사용하게 되면 20초는 1,722프레임이며 1초는 20을 나누어준 86.1프레임이다. 이러한 계산으로 원하는 loop size를 프레임 단위로 입력하여 준다. pfft~오브젝트에서 원하는 속도로 사운드 시그널을 출력하기 위하여 phasor~오브젝트에 playback rate 값을 입력한다. 양수를 입력하면 정방향으로 재생되며, 음수를 입력하면 반대방향으로 반전되어 재생된다. 실시간 phase vocoder를 구현하기 위하여 녹음 시간 조정에 해당하는 패치를 추가하였다. phase vocoder는 사운드 녹음 버튼이 눌리지면 녹음이 되고, 꺼지면 녹음된 부분이 계속 재생된다. 배우의 목소리를 1~2초 단위로 실시간 반전하여 느리게 재생시켜주기 위해선, 지속적으로 녹음버튼의 전원을 눌러주어야 한다. metro~오브젝트<sup>52)</sup>를 사용하여 규칙적으로 녹음버튼을 켜졌다, 꺼짐으로써 1~2초 단위로 대사를 분절하여 프로세싱 해주었다.<sup>53)</sup> reverse와 느린 재생속도의 프로세싱을 통하여 기존의 말소리와 다른 초현실적인 분위기의 사운드를 디자인하였다.

---

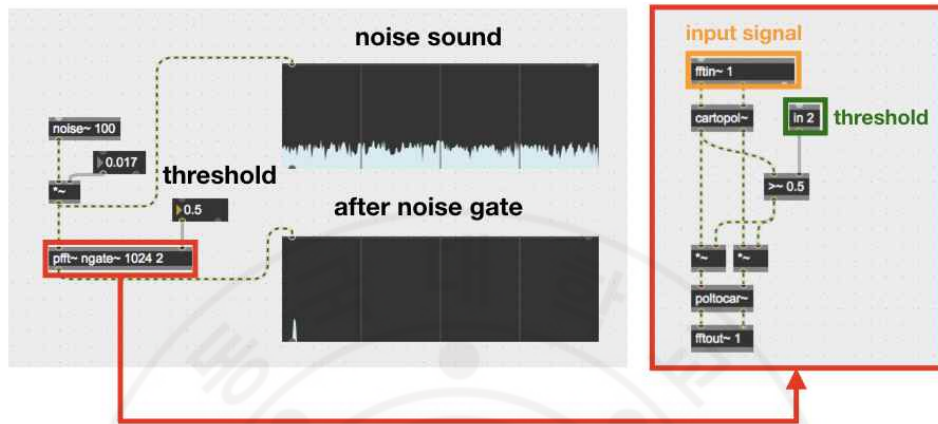
50) 기본적인 골격 구조의 단위

51) 초당 샘플 전송 비율

52) Max에서 일정한 시간 간격으로 bang을 내보내는 오브젝트

53) 강현우 인도음악 연구를 통한 인터랙티브 멀티미디어음악 제작 연구 멀티미디어음악 작품 「<DRAWING DOWN THE MOON>을 중심으로」, 동국대학교 영상대학원: 멀티미디어학과, 석사학위논문, 2017, p. 22

⑦ FFT 분석을 이용한 pfft~ noise~오브젝트 활용



[그림-18] noise gate 패치

배우는 공연 중 말소리 외에도 작은 소음이 발생할 수 있다. 이렇게 수음된 소음을 제거해주기 위해서 pfft~ ngate~오브젝트를 통해 Max에서 노이즈 게이트(noise gate)<sup>54</sup>를 구현하였다. [그림-18]은 noise~오브젝트<sup>55</sup>를 이용하여 극장에서 발생할 수 있는 작은 소음을 발생시켜 노이즈 게이트에 입력시켰다. pfft~ ngate~오브젝트 패치 내부를 살펴보면, cartopol~오브젝트를 거친 사운드 시그널은 >~오브젝트<sup>56</sup>를 통해 각 주파수에 따른 특정 amp 값과 비교되어, 특정 음량 값보다 높으면 통과되며, 낮으면 차단된다. 즉, threshold<sup>57</sup>를 지정하여 특정 음량 값 이하의 사운드 시그널을 차단할 수 있는 것이다. 오브젝트를 거친 after noise gate의 파형을 보면 대부분의 소음이 사라졌음을 알 수 있다.

54) 소음을 차단하는 게이트

55) Max에서 화이트 노이즈를 생성하는 오브젝트

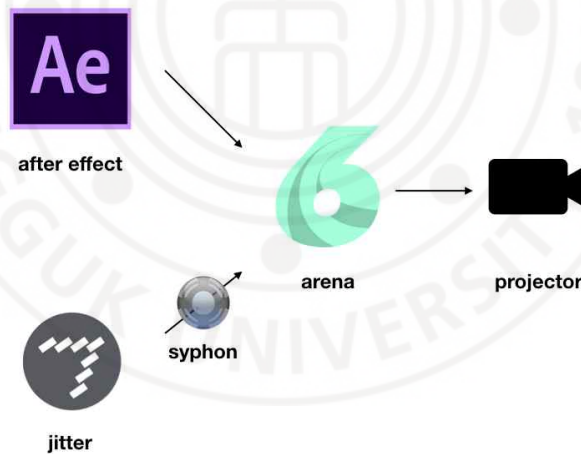
56) Max에서 두 개의 시그널 값을 비교하는 오브젝트

57) 임계지점, 기준의 최대치

## 2. 영상 시스템 연구

### 1) 영상 시스템

사운드와 영상 인터랙션을 구현하기 위해서는 사운드의 데이터 값에 영상이 반응해야 한다. 본 연구에서는 tape music과 목소리의 특정 주파수 음량 값에 따른 영상 인터랙션을 연구하였다. 각기 다른 사운드 소스에 영상을 연동하여 대응되는 이미지를 구현하였다. After Effect<sup>58)</sup>와 Jitter<sup>59)</sup>를 이용하여 영상 이미지를 제작한 후, Syphon<sup>60)</sup>을 통해 Arena<sup>61)</sup>를 거쳐 빔 프로젝터(beam projector)로 최종 출력하는 시스템을 설계하였다. [그림-19]는 영상 시스템을 도식화한 그림이다.



[그림-19] 영상 시스템

58) Adobe사에서 개발한 디지털 모션 그래픽 제작 및 합성 소프트웨어이다.

59) Max에 내장된 어플리케이션이다. 각 grid에 따른 matrix 정보를 통해 영상을 제작한다.

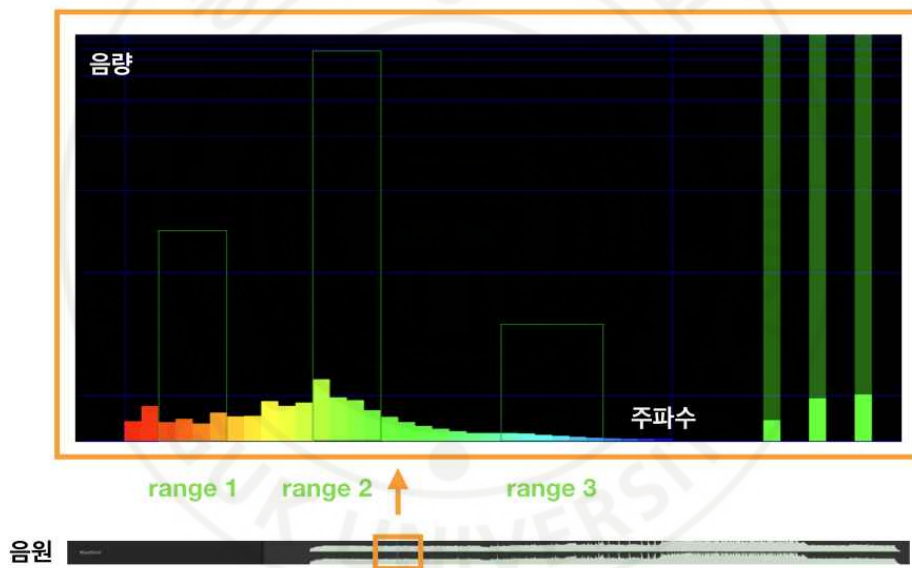
OpenGL, 모션 트래킹, 오디오 비주얼라이제이션 등의 그래픽 효과를 구현할 수 있다.

60) 어플리케이션 간의 영상 프레임을 공유할 수 있는 오픈소스이다.

61) 영상을 제어하는 VJ 프로그램으로 레이어 기능과 실시간 연산 기능이 탁월하다.

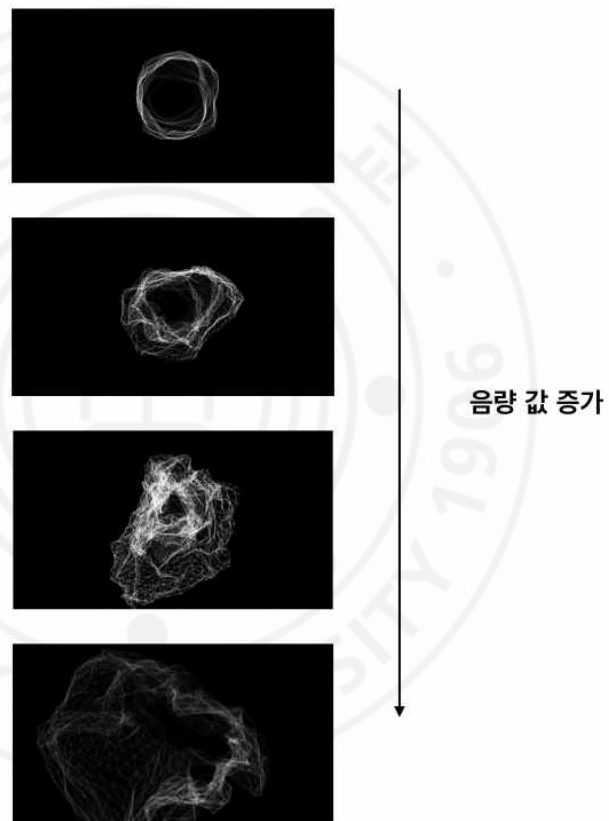
## 2) After Effect를 이용한 영상 제작

After Effect를 활용하여 tape music과 연동되는 영상을 미리 제작하였다. 연극의 구성에 따라 영상 시나리오를 구축한 뒤, 프로그램 안에 제작된 음원 파일을 불러왔다. 음원과 영상의 인터랙션을 위하여 불러온 음원의 스펙트럼을 분석하여 범위를 지정하였다. [그림-20]은 불러온 파형의 주파수 구역(range)을 나누어준 그림이다.



[그림-20] 음원의 주파수 스펙트럼

tape music의 음원을 저음역대, 중음역대, 고음역대로 나뉜 뒤, 각 주파수 구역의 음량 값을 일정한 비율로 확대/축소하여 영상 파라미터에 연동하였다. 이를 통해 살아있는 영원의 축소와 확대를 표현하였다. [그림-21]은 음원의 음량 값에 따라 반응하는 이미지를 나타낸다.

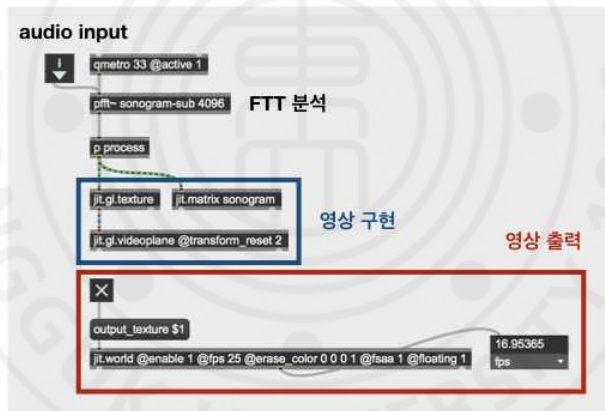


[그림-21] 영상 인터랙션 구현



### 3) Max를 이용한 영상 제작

Max에는 영상 시스템을 구현할 수 있는 Jitter 기능이 있는데, 이를 통해 sonogram<sup>62)</sup>을 구현하였다. tape music 음원을 pfft~오브젝트를 거쳐 주파수 분석을 한 뒤, jit.gl 오브젝트<sup>63)</sup>를 사용하여 제작하였다. 제작된 영상은 jit.world오브젝트<sup>64)</sup>를 거쳐 컴퓨터 화면에 나타난다. [그림-22]는 Max에서 구현한 sonogram 패치이며, [그림-23]은 음원에 연동하여 구현된 영상을 나타낸다. X축은 주파수, Y축은 시간이며, 밝기 값은 주파수 대역에 따른 음량 값이다.

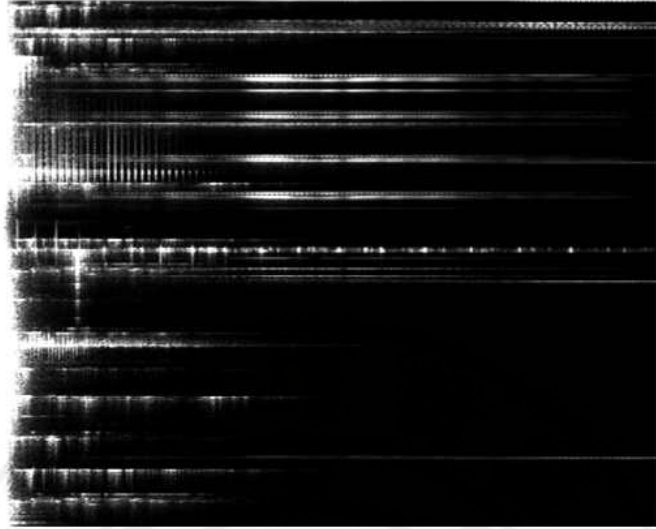


[그림-22] sonogram 패치

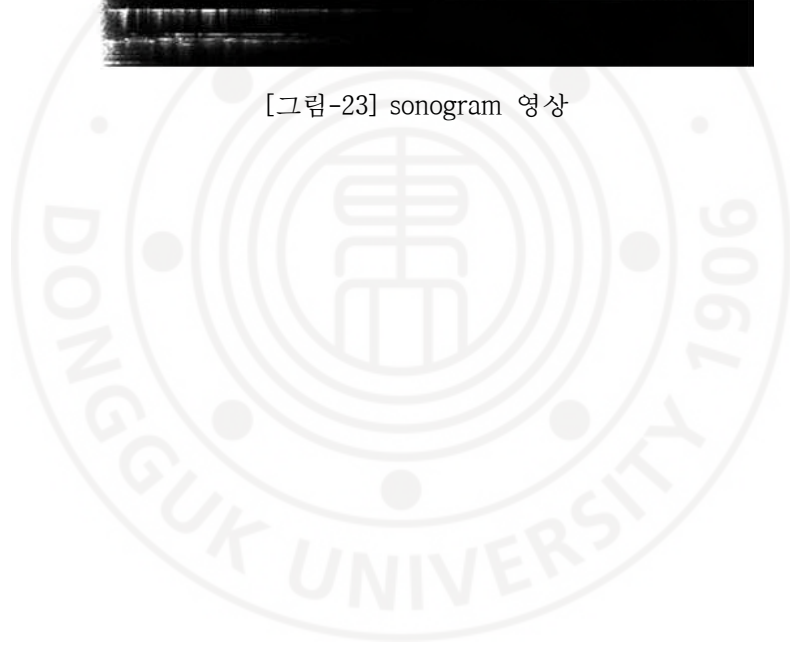
62) 주파수에 따른 음량 값을 시간의 진행에 따라 시각화한 영상

63) Max에서 OpenGL을 제어하는 오브젝트

64) Max에서 영상을 윈도우로 출력하는 오브젝트



[그림-23] sonogram 영상



#### 4) Arena를 이용한 영상 시스템

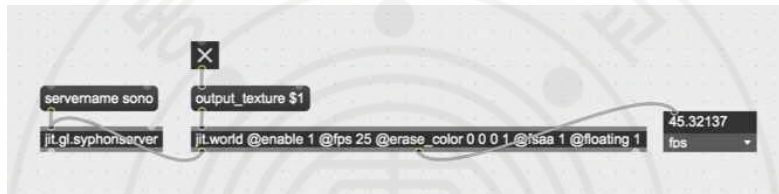
After Effect와 Max로 만들어진 영상을 최종출력해 주기 위해 영상 소프트웨어인 Arena를 사용하였다. Max에서 만든 영상은 Syphon을 이용하여 전송해주었으며, After Effect에서 만든 영상은 파일을 통해 불러왔다. 영상들을 재배치하거나 겹치는 효과를 적용할 수 있으며, 각 장면에 따른 영상효과 및 시나리오를 구현할 수 있다. 또한 프로그램에서 제공하는 기본적인 효과(effects)로 극적인 효과를 보강해줄 수 있다. <표-4>는 작품에 사용된 Arena 내장 효과들에 대한 설명이다.

<표-4> Arena 영상효과

영상효과	효과 설명
transform	영상의 위치와 크기 각도를 변형시키는 효과
edge detection	영상의 경계선을 강조하는 효과
opacity	영상의 투명도를 조절해주는 효과
goo	영상을 울렁이게 하는 효과
blur	영상을 흐리거나 번지게 하는 효과
radialblur	영상의 중심점을 기준으로 번지게 하는 효과
twisted	영상을 뒤틀는 효과
suckr	영상을 빨려 들어가게 하는 효과
flip	영상을 수직 및 수평으로 뒤집는 효과
trails	영상의 잔향을 조절해주는 효과

## 5) Syphon을 이용한 영상 데이터 전송

[그림-24]는 jit.gl.syphonserver 오브젝트를 사용하여 Max에서 제작한 영상 프레임을 Arena로 보내기 위한 패치이다. 메시지 박스<sup>65)</sup>에 이름을 지정하여 서버를 열어주고선 jit.world 오브젝트에서 출력된 영상 프레임들을 jit.gl.syphonserver 오브젝트로 입력시키면 서버에 공유된다. [그림-25]는 공유된 프레임이 Arena를 거쳐 최종 출력된 영상이다.



[그림-24] Syphon 전송 패치



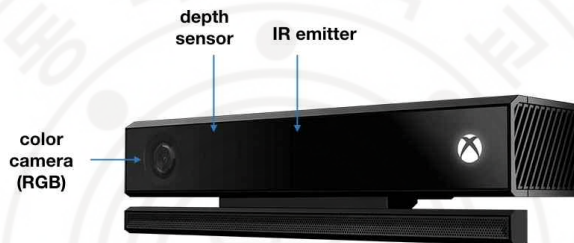
[그림-25] Arena 최종 출력

65) Max에서 symbol, list 등의 데이터를 전송하거나 나타내는 오브젝트

### 3. 동작인식 시스템 연구

#### 1) 모션 트래킹 데이터 추출

모션 트래킹(motion tracking)<sup>66)</sup> 센서의 한 종류인 Kinect는 RGB 컬러 카메라, IR emitter, depth sensor로 구성되어있다.



[그림-26] Kinect 구성도

좌측에 위치한 IR emitter에서 픽셀 단위의 무수한 점을 방사하게 되면 피사체에 반사된 값을 우측에 depth sensor에서 받아 인식한다. 그렇게 수신된 픽셀의 거리 값으로 X, Y, Z의 3차원 좌표를 만들어 인식한다.

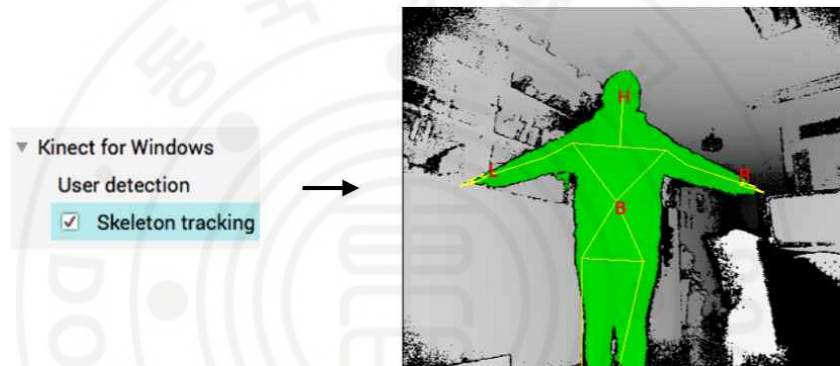
Kinect는 별도의 USB 케이블을 컴퓨터에 연결하여 데이터를 전달할 수 있다. 본 연구는 macOS 기반의 환경에서 센서 데이터를 추출 할 수 있는 소프트웨어인 NI Mate<sup>67)</sup>를 사용하여 진행하였다.

66) 동작을 인식하여 데이터를 출력하는 기법

67) Kinect의 정보데이터를 컴퓨터로 전달받을 수 있는 소프트웨어

## 2) 모션 트래킹 데이터 매핑

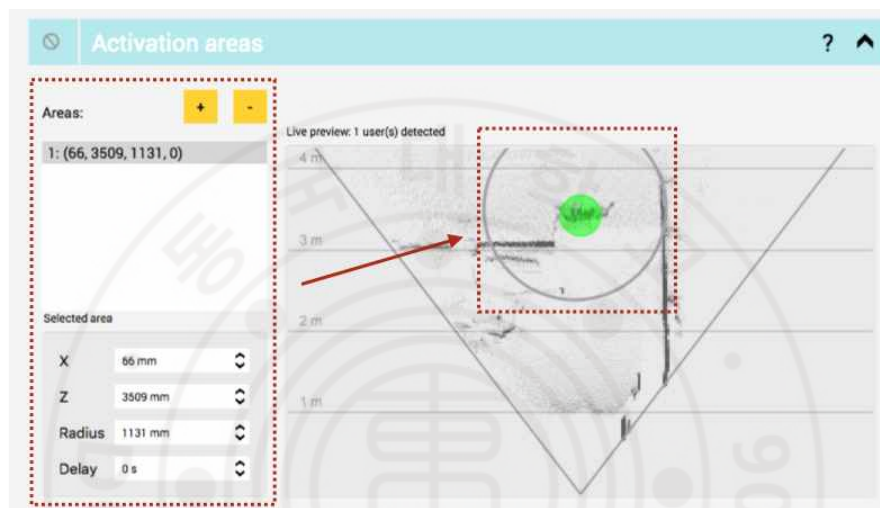
Kinect를 통해 배우의 움직임을 감지하고, NI-mate를 사용하여 skeleton data를 추출한다. skeleton data는 촬영된 신체 구조를 통하여 각 관절의 움직임을 수치로 환산한 것이다. 이 데이터는 OSC(open sound control)<sup>68)</sup> 통신을 통해 Max로 전송된다. [그림-27]과 같이 skeleton tracking 버튼을 활성화하여 데이터를 추출할 수 있다.



[그림-27] skeleton tracking 이미지

68) Open Sound Control, 음악적인 용도를 위해 제작된 네트워크 프로토콜이다.

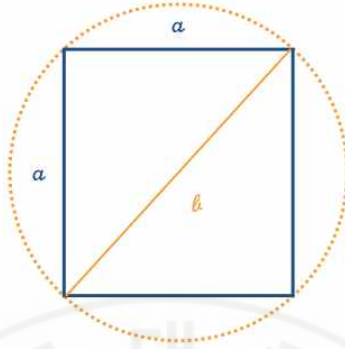
극장에서 배우는 Kinect와 2,500mm(밀리미터) 떨어진 지점에 있는 네모난 직사각형 인터페이스 내에서 움직인다. [그림-28]은 NI-mate에서 모션 트래킹을 활성화시킬 영역을 설정해주는 그림이다.



[그림-28] NI-mate에서의 영역 설정

Area를 통하여 구역을 설정해줄 수 있으며, X와 Z 파라미터를 통하여 영역의 중심점 위치를 지정해 줄 수 있다. radius 값은 영역을 지정하는 원의 반경을 의미한다. 극장에서 배우는 한 변이 1,600mm인 정사각형 내에서 움직이기에 원의 Area 면적을 사각형 영역보다 넓게 설정해주어야 한다. [그림-29]와 같이 피타고라스<sup>69)</sup> 계산식을 이용하여 1,131mm란 결과 값이 나왔다. 결과 값은 radius에 입력되어 배우의 동선을 포함하는 영역을 설정하였다.

69) 직각삼각형에서 빗변 길이의 제곱은 다른 두 변의 길이의 제곱의 합과 같다.



$$a^2 + a^2 = b^2$$

[그림-29] 피타고라스 공식

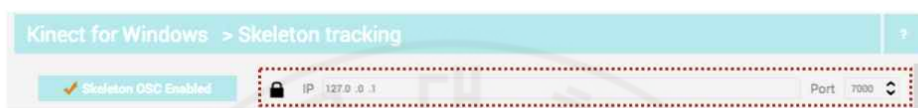
[그림-30]의 format은 트래킹한 데이터를 어떠한 형태로 전송할지를 정한다.  
 데이터를 보내기 전, 원하는 종류에 따라 통신 명칭의 변경이 가능하다.



[그림-30] NI-mate에서의 format 설정



NI-mate에서 트래킹한 데이터는 OSC 통신을 통해 Max로 보내야 하는데, 이를 위하여 IP<sup>70)</sup>와 port<sup>71)</sup> 번호를 지정해줘야 한다 [그림-31]. 동일한 컴퓨터 내부에서 데이터가 전송되기에 local host<sup>72)</sup> IP 주소인 127.0.0.1을 입력하였다. 양쪽의 소프트웨어에서 port 번호를 일치시켜야 통신이 가능하다.



[그림-31] NI-mate에서의 IP와 port 설정

배우의 움직임 데이터는 NI-mate를 거쳐 OSC 데이터로 Max에 보내어진다. udpreceive오브젝트는 다른 소프트웨어에서 보내온 OSC 데이터를 받을 수 있는 Max 오브젝트이다. [그림-32]는 udpreceive오브젝트로 전달된 데이터를 route오브젝트<sup>73)</sup>와 unpack오브젝트<sup>74)</sup>를 사용하여 머리, 몸, 왼손, 오른손 데이터 값의 X, Y, Z 값을 추출한 패치이다.

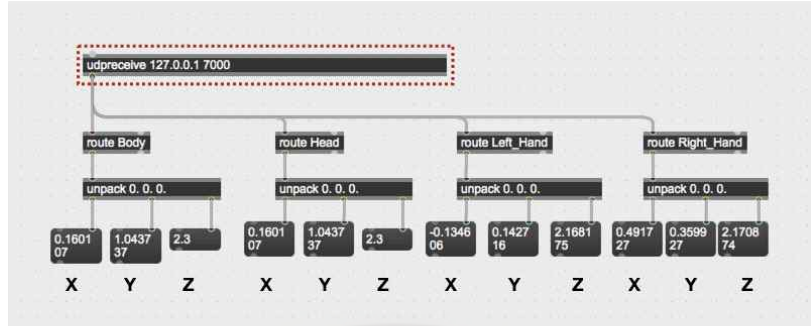
70) 프로토콜에 따라 네트워킹이 가능한 장비를 식별하는 주소이다.

71) 운영 체제 통신의 중단점이다.

72) 컴퓨터 네트워크에서 사용하는 루프백 호스트명으로, 자신의 컴퓨터를 의미한다.

73) Max에서 데이터를 구분하여 분류하는 오브젝트

74) Max에서 묶인 데이터를 개별로 풀어내는 오브젝트



[그림-32] OSC 데이터 값 추출 패치

모션 센서로 추출된 데이터를 매핑하기 위해서는 배우의 신체적인 움직임에 따른 안정적인 트래킹 여부가 중요하다. 인터페이스 내에서 움직이는 동작 범위는 Kinect 투사 범위를 벗어나지 않아 문제가 되지 않았지만, 배우의 안무에 따라 양쪽 팔에 대한 트래킹은 불안정한 상태를 보였다. <표-5>는 배우의 행위에 따른 신체 트래킹 안정성에 대한 실험 결과를 나타낸다.

<표-5> 동작에 따른 트래킹 안정성 실험 결과

동작	각각의 신체 데이터에 따른 안정성			
	머리	몸통	왼손	오른손
한 손을 뻗는다	좋음	좋음	좋음	좋음
양손을 뻗는다	보통	좋음	나쁨	나쁨
머리를 움켜쥔다	좋음	좋음	나쁨	나쁨
날갯짓을 형상화한다	좋음	좋음	나쁨	나쁨
웅크리며 주저앉는다	보통	좋음	나쁨	나쁨
옷깃에 이끌린다	좋음	좋음	보통	보통

## 4. 인터페이스 PRIZONE 제작 연구

인터페이스 PRIZONE은 극 중에서 삶과 죽음의 경계선에 대한 오브제임과 동시에 배우의 내면을 시각화하고자 제작한 인터페이스이다. PRIZONE은 4개의 LED에 배우의 연기가 연동됨으로써 시각적, 음악적, 극적인 효과를 동시에 일으킨다. 이를 제어하기 위해선 디지털 신호를 전달받으며, 아날로그 신호를 동시에 발생시키기 위한 micro controller<sup>75)</sup>가 필요했다. 이러한 이유에서 Arduino UNO<sup>76)</sup>를 사용하여 인터페이스 PRIZONE을 구현하였다.

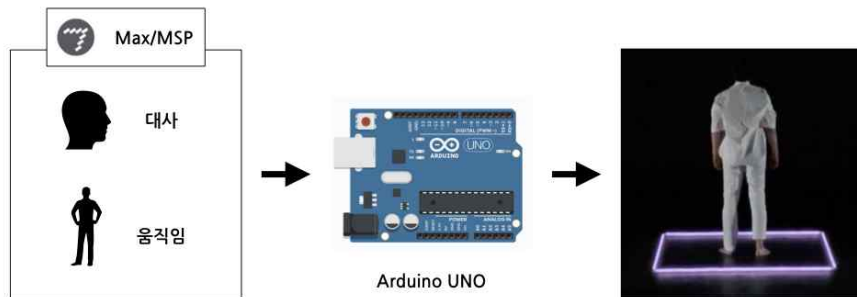
### 1) Arduino를 이용한 인터페이스 시스템 구축

Arduino는 micro controller board를 기반으로 제작된 인터랙티브한 객체들과 디지털 장치를 만들기 위한 도구이자 소프트웨어 개발 환경이다. 본 연구에서는 Arduino UNO에 USB 단자를 컴퓨터와 연결하여 이용하였다. 움직임 데이터와 목소리 음량 값을 시각화하기 위해서 Max로부터 각각의 데이터 값을 Arduino로 입력해주어야 한다. [그림-33]은 PRIZONE의 연동 시스템이다.

---

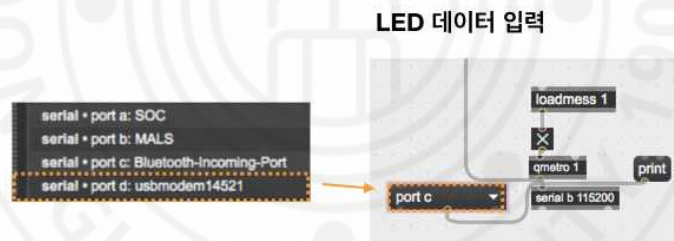
75) 마이크로 프로세서와 메모리, 입력장치 등의 필요한 기능이 함께 집적된 장치이다. 센서나 LED와 같은 장치들을 연결하여 사용할 수 있다.

76) Microchip ATmega328P 마이크로 컨트롤러를 기반으로 하며 Arduino.cc가 개발한 오픈 소스 마이크로 컨트롤러 보드이다.



[그림-33] PRIZONE 인터페이스 연동 시스템

Max에서 전달받은 데이터를 Arduino로 전달해주기 위해 serial 통신<sup>77)</sup>을 이용하였다. [그림-34]는 Max에서 serial오브젝트를 이용하여 LED 밝기 값에 대한 데이터를 전달하는 패치이다.



[그림-34] serial 통신 구현 패치

77) 직렬통신이며, 다른 기기와 데이터를 주고받기 위한 통신 방법에 해당한다.

serial 통신을 구현하기 위해서는 각 프로그램의 통신 속도를 일치시켜줘야 한다. serial오브젝트에 연결된 print 메시지 박스를 누른 후 Max console에서 usbmodem에 해당하는 port를 열어준다. LED의 밝기 데이터는 4개의 디지털 핀에 매핑되어 아날로그 전기신호로 전환된다. [그림-35]는 아두이노 통합개발환경(IDE)<sup>78)</sup>을 활용하여 인터페이스 시스템을 코딩한 화면이다.



```

LEDCONTROL $

int LED_pinA = 3; //LED_pinA 변수지정, 3번 데이터핀
int LED_pinB = 5; //LED_pinB 변수지정, 5번 데이터핀
int LED_pinC = 6; //LED_pinC 변수지정, 6번 데이터핀
int LED_pinD = 11; //LED_pinD 변수지정, 11번 데이터핀

void setup()
{
  Serial.begin(115200); //시리얼 통신 시작, 통신 속도 115200 설정
  pinMode(LED_pinA, OUTPUT); //각 디지털핀을 출력으로 설정
  pinMode(LED_pinB, OUTPUT);
  pinMode(LED_pinC, OUTPUT);
  pinMode(LED_pinD, OUTPUT);
}

void loop(){
  while (Serial.available()) { //수신 데이터중 읽지 않은 데이터 반환
    int value1 = Serial.parseInt(); //value 변수 값에 LED 밝기 값 대입
    int value2 = Serial.parseInt();
    int value3 = Serial.parseInt();
    int value4 = Serial.parseInt();

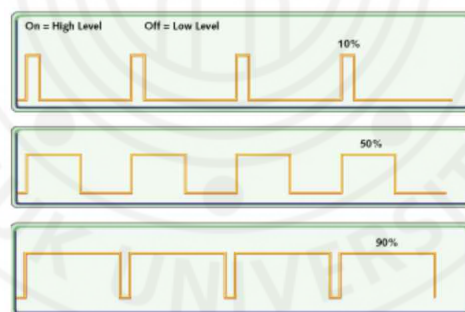
    if (Serial.read()){ //수신 데이터 중 저장된 순서대로 변환
      analogWrite(LED_pinA, value1); //LED핀을 밝혀준다
      analogWrite(LED_pinB, value2);
      analogWrite(LED_pinC, value3);
      analogWrite(LED_pinD, value4);
    }
  }
}

```

[그림-35] Arduino IDE를 활용한 LED 제어 코딩

78) 프로그램을 작성할 수 있게 도와주는 도구이며, 작성한 프로그램을 컴파일 하여 Arduino 보드에 업로드 하는 작업을 도와주는 스케치 프로그램이다.

Arduino의 디지털 핀 중 ~가 붙은 핀은 PWM(펄스 폭 변조)이 가능한 핀으로 디지털 신호로 아날로그 신호와 같은 효과를 줄 수 있다. PWM은 디지털 신호인 0과 1의 비율을 조절하여 LED 밝기 값을 조절한다. 1은 전원을 ON 시키며 0은 전원을 OFF 시킨다. [그림-36]<sup>79)</sup>은 서로 다른 세 가지의 PWM을 나타낸다. 상단의 그림은 duty cycle<sup>80)</sup>이 10%일 때의 PWM 출력을 나타낸다. 즉, 주기의 10% 동안 1이라는 신호를 내보내고, 나머지 90% 동안은 0이라는 신호를 내보낸다. 그리고 중간 하단은 각각 50% 90% 일 때의 PWM 출력을 보여준다. 이러한 세 가지 duty cycle이 다른 PWM 출력은 다른 아날로그 신호로 인코딩(encoding)된다. 예를 들어, 9V의 전원을 공급하고 duty cycle이 10%라면 0.9V의 아날로그 신호가 인코딩된다. PRIZONE은 5V의 전원이 공급되기에 duty cycle이 10%라면 0.5v가 출력된다. 이러한 원리를 통하여 LED의 밝기를 조절해주었다.



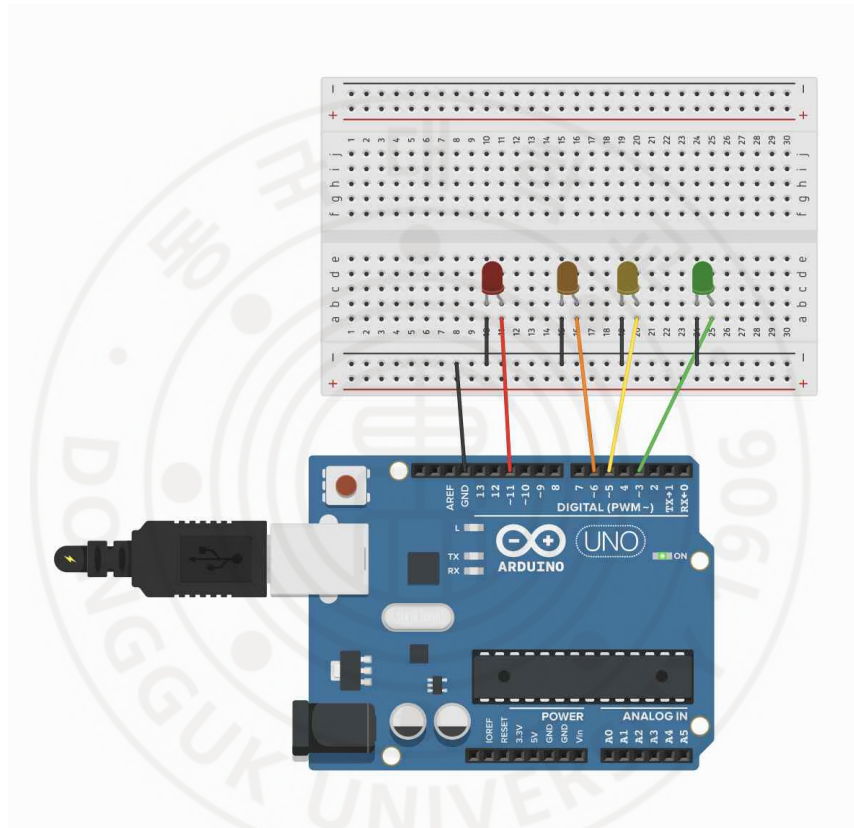
[그림-36] duty cycle에 따른 PWM

79) Michael Barr, "Introduction to Pulse Width Modulation(PWM)" (O'Reilly Network,2003)

참조

80) 반복되는 주기

[그림-37]은 브레드보드(breadboard)<sup>81)</sup>를 이용하여 Arduino와 4개의 LED를 연결한 도면이다. Arduino의 GND<sup>82)</sup>핀은 LED의 음극에 연결해주었으며, 디지털 핀은 각 LED의 양극에 연결해주었다.



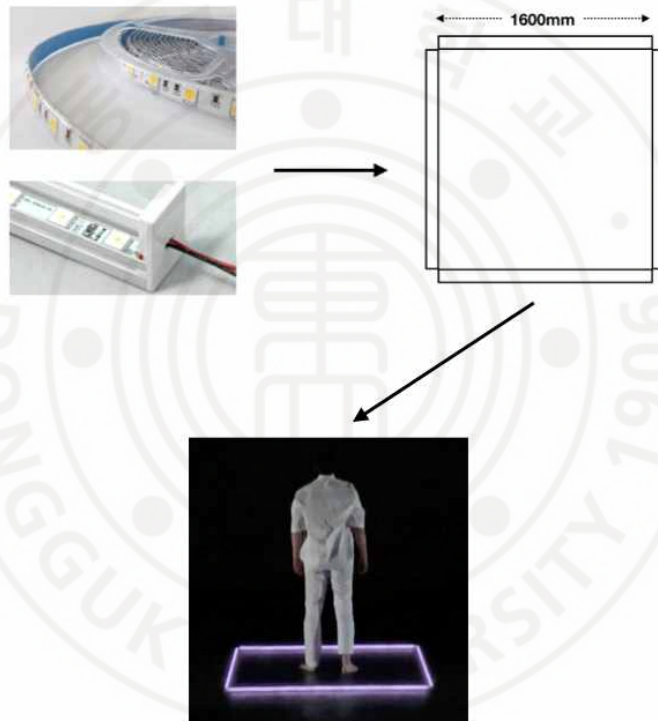
[그림-37] 브레드보드를 이용한 Arduino와 LED 연결 도면

81) 전기 및 전자회로 실험에서 기판에 납땀을 하지 않고도 회로를 구성할 수 있는 도구이다.

82) 접지 또는 그라운드는 전기 회로나 전기 기기 따위를 도체로 땅에 연결하는 것을 말한다.

## 2) PRIZONE의 Interactive Visualization을 위한 LED

PRIZONE은 총 4개의 LED를 [그림-38]과 같이 설계하였다. 5V 5050 3chip LED strip을 1,600mm씩 재단하여 알루미늄으로 제작된 방열판에 부착하여, 반투명한 커버를 입혀주어 빛의 확산이 이루어지도록 제작하였다.

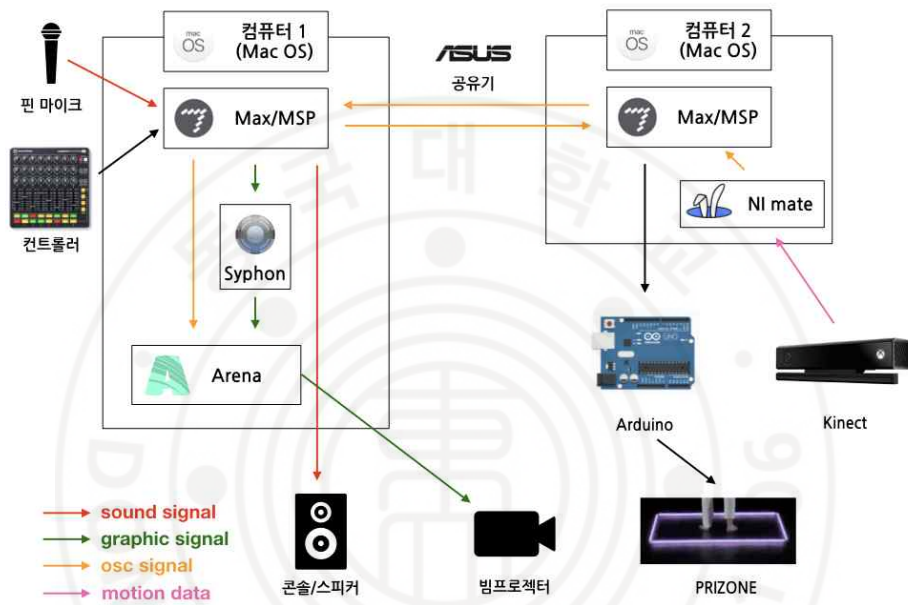


[그림-38] PRIZONE 하드웨어 제작과정



## 5. 실시간 공연 시스템 연구

### 1) 공연 시스템



[그림-39] 공연 시스템 설계도

[그림-39]는 무대에서 사용된 총 시스템에 대한 설계도이다. 무대에 위치한 Kinect와 PRIZONE을 제어하기 위하여 무대 쪽에 컴퓨터 한 대를 설치하였다. 콘솔 쪽에 위치한 컴퓨터와의 통신을 위해 무선 공유기를 이용하여 컴퓨터 간의 OSC 데이터를 주고받았다. 목소리는 모노 채널로 오디오 인터페이스를 통해 입력받았으며, Max에서 사운드 프로세싱을 거쳐 스피커로 출력하였다. 영상은 Max에서 Syphon을 통해 전달받았으며, Arena에서 영상효과를 입혀주었다. 배우의 움직임 데이터와 음량 값은 PRIZONE의 LED 밝기에 연동되었다.

## 2) 실시간 인터랙션 시스템 구축

### ① 사운드 인터랙션 시스템 연구

배우의 음성은 사운드 프로세싱 파라미터 값에 움직임 데이터가 연동되어 이루어졌다. <표-5>의 결과에 따라 비교적 안정적인 몸통과 머리를 트래킹한 데이터를 사용하기로 하였다. Kinect 데이터를 매핑하기 위하여 움직임의 X,Y,Z축에 따른 범위에 대한 수치를 실험을 통해 알아냈다. <표-6>과 <표-7>은 움직임에 따른 몸통과 머리의 최대, 최소 값을 정리한 표이다.

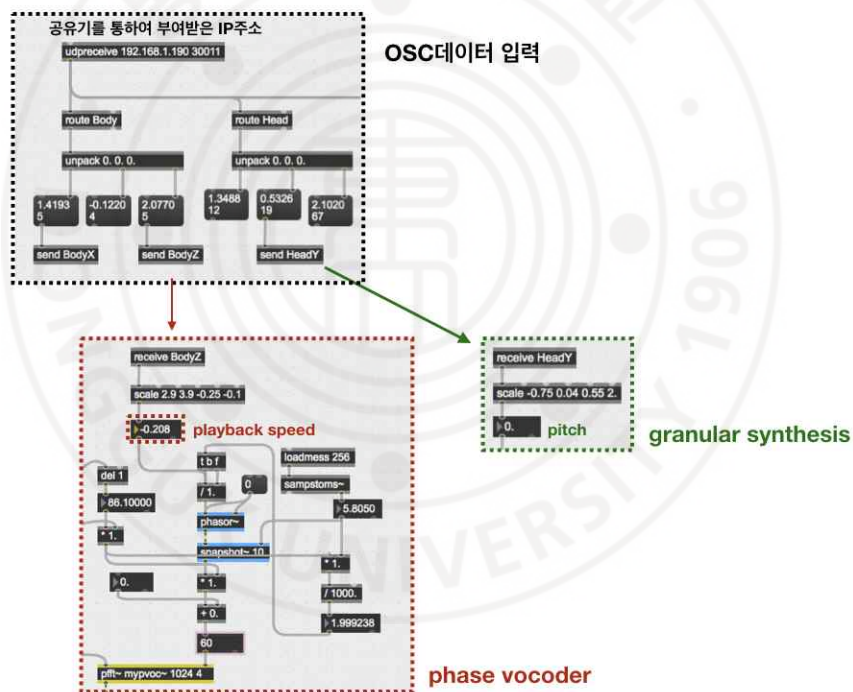
<표-6> 머리와 몸통의 최대, 최소 위치 값

	몸통		머리
	X	Z	Y
최대	0.9	3.9	0.04
최소	-0.6	2.9	-0.75

<표-7> 머리와 몸통의 최대, 최소에 따른 방향

	몸통		머리
	X	Z	Y
최대	무대 좌측	무대 뒤쪽	상승
최소	무대 우측	무대 앞쪽	하강

udpreceive오브젝트를 이용하여 전송받은 데이터는 Max의 사운드 프로세싱 패치로 전송되어 granular synthesis의 grain pitch 값과 phase vocoder의 playback speed를 조절해준다 [그림-40]. 배우의 상승과 하강에 대한 움직임은 머리의 Y축에 해당하는 데이터로 매핑하였다. 배우가 몸을 일으켜 세움에 따라 grain pitch 또한 높아지며 절망 속에서 희망을 향해 날아가는 느낌을 주었다. playback speed 같은 경우, 몸통의 Z값을 이용하여 배우가 관객석에서 멀어지면 재생속도를 늦춰주는 역할을 하였다. 이때 scale오브젝트<sup>83)</sup>를 활용하여 범위 값을 -0.25 ~ -0.1로 치환하여 미세한 차이로 속도가 바뀌는 효과를 연출하였다.

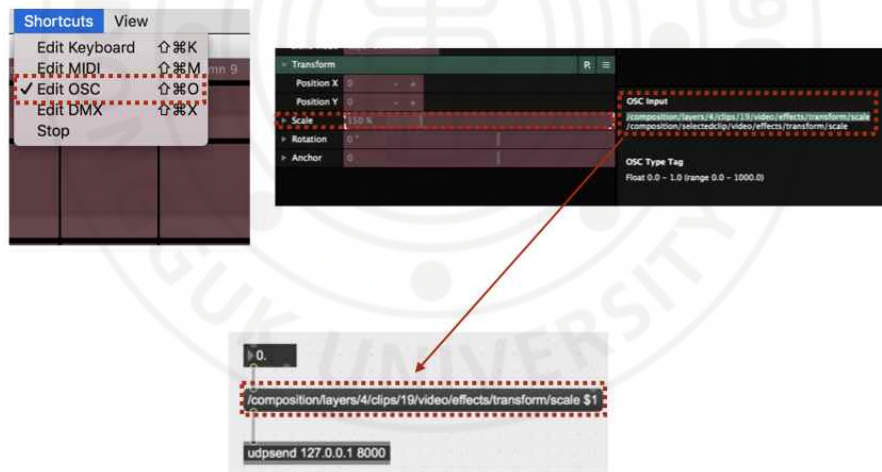


[그림-40] udpreceive오브젝트를 이용한 사운드 인터랙션 패치

83) Max에서 치환을 통하여 원하는 범위를 지정할 수 있는 오브젝트이다.

## ② 영상 인터랙션 시스템 연구

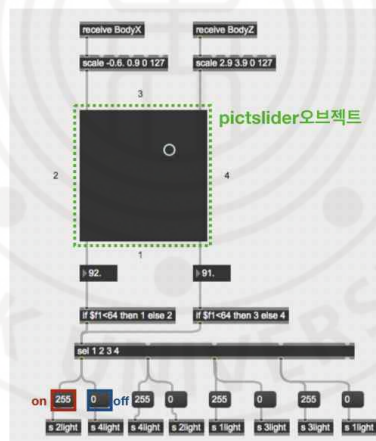
영상은 프로세싱된 음성에 반응하여 변화하였다. Arena의 파라미터에 목소리 음량 값을 매핑하여 다양한 영상효과를 주었다. 이를 위하여 OSC 통신을 사용하였는데, Max에서 Arena로 데이터를 전송해주기 위해선 특정 파라미터에 대한 주소를 입력해주어야 한다. [그림-41]은 Arena의 특정 파라미터에 대한 OSC 주소 정보를 활용하여 Max에 입력하는 과정을 보여준다. Edit OSC를 클릭하면 지정할 수 있는 OSC 영역이 빨간색으로 변한다. OSC input의 상단에 위치한 주소를 복사하여 Max의 udpsend 오브젝트와 연결한 뒤, port를 맞춰주고 OSC 데이터 전송해주면 된다. OSC Type Tag를 통해 작동하는 데이터 종류와 범위를 알 수 있다.



[그림-41] Arena와 Max의 통신을 위한 OSC 주소 설정

### ③ 인터페이스 인터랙션 시스템 연구

PRIZONE은 장면에 따라 목소리 음량 값과 움직임 값에 의해서 반응하도록 설계하였다. 두 가지 데이터 모두 OSC 통신과 serial 통신을 이용하였다. LED는 각각 1~4번으로 번호를 지정하여 밝기 값을 조절해 주었다. 배우가 PRIZONE을 벗어나기 위한 연기를 하면, 움직이는 방향에 따라 벽이 생겨나는 연출을 하였다. [그림-42]는 배우의 위치 데이터를 pictslider오브젝트<sup>84)</sup>에 매핑하여 LED를 제어하는 Max 패치이다. pictslider오브젝트 X축, Y축은 좌측아래 지점을 기준으로 0~127의 범위를 지정한다. Kinect를 통해 추출된 데이터는 scale오브젝트를 통하여 0~127의 범위로 치환된다. 그 후 if오브젝트<sup>85)</sup>를 이용하여 알고리즘을 만들어주어 LED 자동화 시스템을 구축하였다.



[그림-42] PRIZONE 인터랙션 패치

84) Max에서 사각형의 X축과 Y축에 매핑하여 사용할 수 있는 오브젝트이다.  
85) Max에서 if 가정문의 코딩을 구현할 수 있는 오브젝트이다.

### Ⅲ. 연구 기술의 작품 적용

연구된 tape music 사운드 디자인과 목소리의 실시간 사운드 프로세싱을 이용한 멀티미디어 퍼포먼스 <Eternity>는 2019년 11월 16일 동국대학교 이해량 예술극장에서 진행된 ‘SEEING SOUND LISTENING IMAGE(보는 소리, 듣는 영상 XVI)’ 공연에 초연되었다.

#### 1. 작품소개

<Eternity>는 주인공이 삶과 죽음의 경계선에서 영원을 찾아 헤매는 과정을 연출한 멀티미디어 퍼포먼스 연극이다. 주인공은 영혼과 영원 사이의 간극을 좁히려 하지만, 결국 영원에 의해 잠식되어버리는 결말을 맞이한다. ‘끊임없는 갈망’, ‘고갈된 희망’, ‘절망의 굴레’, ‘진리에 대한 집착’, ‘뜨거운 열망’ 등의 초목표<sup>86)</sup>를 설정하였다. 프랑스 시인 Arthur Rimbaud의 시, <L'Éternité>를 읽고서 받은 영감에 착안하여 대사를 만들었으며, 연의 구분에 따라 장면(scene)을 나눠주었다. 제작한 인터페이스 PRIZONE을 통해 살아 숨 쉬는 초현실적인 공간을 만들었다. 공연 사운드는 Max로 제작한 사운드 합성법과 실시간 사운드 음향효과를 사용하였다. 다양한 뉴미디어를 통해 배우, 사운드, PRIZONE, 영상이 연결되도록 연출했다. 장면별로 프리셋(preset)<sup>87)</sup>을 설정하여 각기 다른 사운드 프로세싱, 영상, PRIZONE 효과를 연출하였다.

---

86) 최종적으로 도달해야 하는 중요한 목표

87) 사전에 설정해둔 값

배우의 대사는 어절 단위로 느리게 반전한 뒤 다른 차원에 대한 존재를 표현하였다. tape music은 전체적으로 저음역대의 베이스 소리로 스산한 느낌을 주었다. 공간의 긴장감을 조성하기 위하여 합성된 사운드의 panning값을 빠르게 변화하여 형체가 움직이는 효과를 주었다. C파트에서부터 나오는 로우 임팩트 사운드(low impact sound)<sup>88)</sup>의 주기를 점점 빠르게 하여 클라이맥스로 고조시켰다. <표-8>은 작품 <Eternity>의 전체적인 구조에 대한 설명이다.

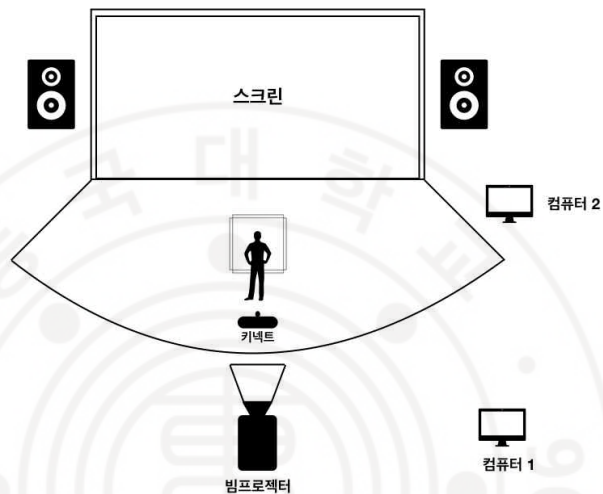
<표-8> 작품 <Eternity> 전체 구조

장면	시간	초목표	사운드 프로세싱	PRIZONE
A파트	0:00~3:10	탐색, 경계	delay chorus	시나리오
A'파트	3:10~4:48	착시, 착란	delay chorus phase vocoder	시나리오
B파트	4:48~5:32	채찍질, 간절함	phase vocoder flanger	음성 인터랙션
C파트	5:32~6:46	감금, 해방에 대한 희망	phase vocoder flanger comb filter granular synthesis	움직임 인터랙션
D파트	6:46~7:58	불타오르는 몸	X	시나리오
A''파트	7:58~9:50	굴레, 영원을 향한 발걸음	phase vocoder comb filter granular synthesis	시나리오

88) 저음의 형태로 어택감이 강하게 제작된 사운드

## 2. 무대 및 영상 구성

### 1) 무대 구성



[그림-43] 무대 시스템 구성

[그림-43]은 무대 구성도이다. 무대 정중앙에 PRIZONE을 배치하였고, Kinect는 인터페이스의 2.5M(미터) 앞에 설치하여 배우의 움직임을 디텍팅 하였다. 프로젝터를 통해 영상은 객석 뒤에서 스크린으로 투영된다. 컴퓨터는 2대를 사용하였다. 컴퓨터 한 대는 객석의 우측 뒤편의 오퍼레이터 석에 위치하여 모니터를 하며 제어하였고, 다른 한 대는 Kinect와 PRIZONE을 연결하기 위하여 무대 상수<sup>89)</sup> 포켓에 위치하였다.

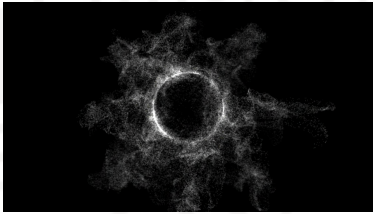
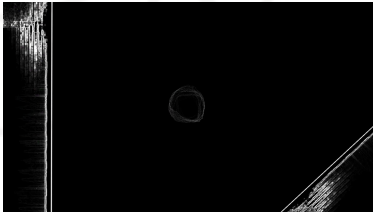
89) 관객이 바라보는 기준으로 무대의 오른편

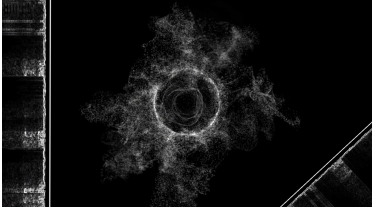
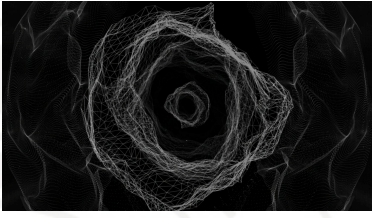
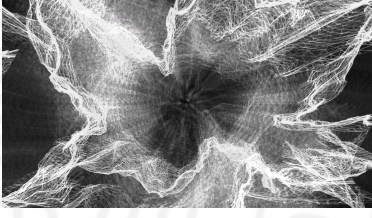
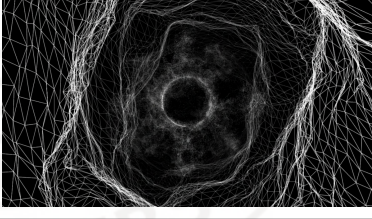


## 2) 영상 구성

영상은 흑과 백의 두 가지 색상만을 이용하여 현실을 넘어서는 분위기를 자아내었다. 주인공의 자아와 처한 공간에 대한 괴리감을 보여주기 위하여, 서로 다른 이미지가 다르게 반응하도록 제작하였다. <표-9>는 장면별 영상의 대표 이미지와 의미를 보여준다. ‘영혼’의 이미지는 ‘영원’의 이미지가 출현함에 따라 끊임없이 간극을 좁혀나가려는 형태를 보인다. 영원은 점점 확장되어 영혼을 분절시키고, 절정에 이르러서는 폭발하여 영혼은 파편이 되어 흩날린다. 결국 영혼의 조각들은 영원의 블랙홀 속에서 소멸한다.

<표-9> 영상 구성

장면 및 의미	이미지
A파트 - 영혼의 들숨과 날숨	
A'파트 - 영원의 끝없는 지평선과 형체의 출몰	

<p>B파트 - 영혼과 영원 사이의 좁혀질 수 없는 간극</p>	
<p>C파트 - 영원의 확장과 영혼의 분리에 대한 고통</p>	
<p>D파트 - 영원의 폭발에 따른 영혼의 재탄생</p>	
<p>A"파트 - 영원에 장식되어 소멸되는 영혼</p>	

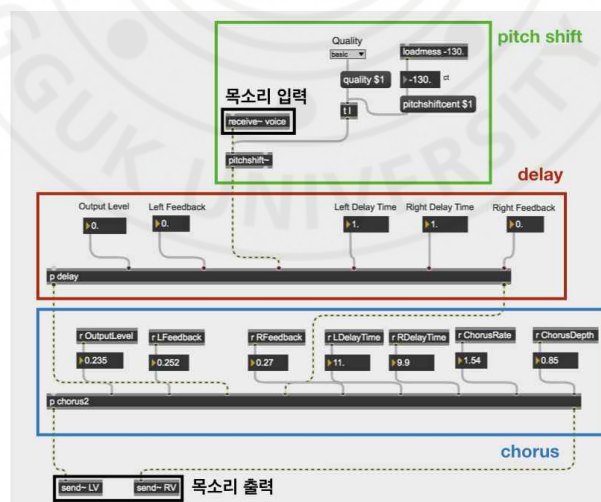
### 3. 연구 기술 적용

#### 1) A파트

〈표-10〉 A파트 구성

시간	0:00 ~ 3:10
사운드 프로세싱	pitch shift + delay + chorus
영상효과	scale opacity blur
PRIZONE 효과	X

A파트에서 배우의 목소리는 [그림-44]와 같이 pitch shift, delay, chorus 사운드 프로세싱을 거쳐 본래의 소리보다 낮은 목소리가 메아리치듯 나와 음침한 분위기를 연출한다.



[그림-44] A파트 사운드 프로세싱 패치

목소리는 pitch shift 음향효과를 거쳐 음정을 낮춰주어, 기존의 배우가 낼 수 없는 저음 배음을 보강하여주었다. 그 후 오른쪽과 왼쪽의 delay time을 다르게 주어 양쪽으로 메아리치듯 나오게 해주었다. 그 후 chorus 음향효과를 거쳐 여러 영혼의 목소리들이 이야기하듯 약간의 변조를 더 해준 뒤 출력하였다.

[그림-45]와 같이 목소리 음량 값에 따라 영혼 이미지는 생성되었다가 소멸한다. After Effect로 제작한 영상은 blur 효과를 거쳐 불투명도(opacity)와 크기(scale)가 변화함에 따라 스크린에 투영되었다. 음량 값이 작아지면 크기는 축소하고 투명해지며, 음량 값이 커지면 크기가 커지고 불투명하게 변화하여 살아 숨 쉬는 영혼의 이미지를 형상화하였다.



[그림-45] A파트 목소리에 따른 영상 인터랙션

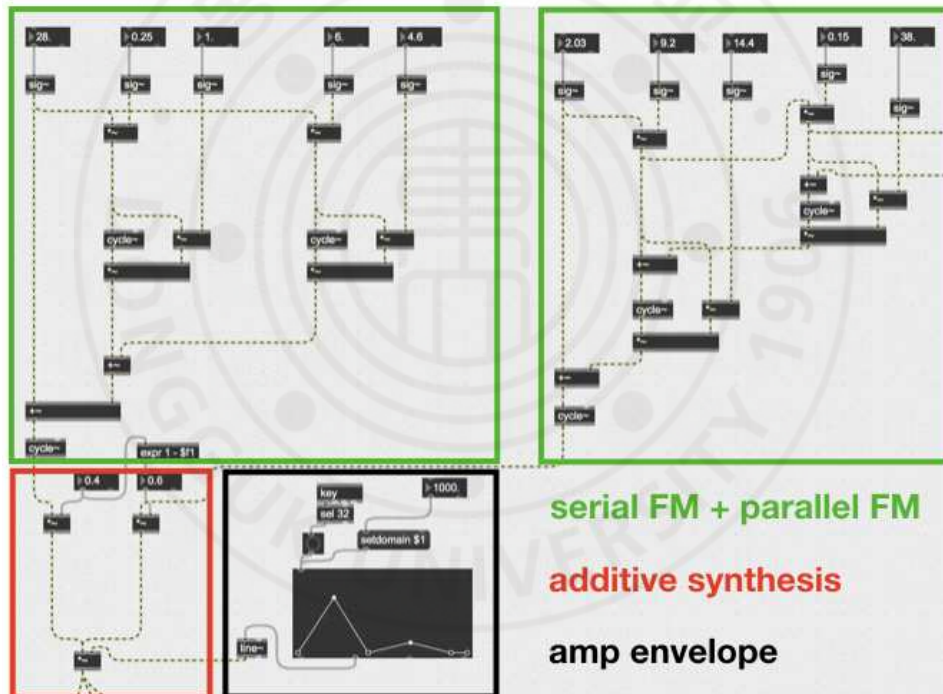
## 2) A'파트

<표-11> A' 파트 구성

시간	3:10 ~ 4:48
사운드 프로세싱	pitch shift + delay + chorus phase vocoder
영상효과	edge detection scale opacity blur
PRIZONE 효과	시나리오

A' 파트에서는 A파트의 사운드 프로세싱에 실시간 phase vocoder 음향효과를 추가하였다. playback speed를 -0.25로 설정하여, 느리게 반전된 배우의 대사가 맞물려 초현실적이며 유영하는 분위기를 냈다.

tape music이 재생되기 시작하며, additive synthesis를 이용하여 제작한 낮은음의 베이스가 무겁고 가라앉는 느낌을 준다. [그림-46]은 중간마다 등장하는 짧은 효과음들을 다양한 FM synthesis 알고리즘과 additive synthesis로 제작한 Max 패치이다.



[그림-46] tape music 소스 제작 패치

소리 합성법을 통해 생성된 사운드는 최종적으로 amp envelope를 거치는데, 짧고 강렬한 사운드를 제작하기 위하여 지속 시간을 1,000ms로 지정해주었다. 또한 attack을 살리기 위하여 function오브젝트를 통해 초기 amp 값을 빠르게 올렸다, 내려주었다. 최종적으로 panning 값을 변화시켜주면서 특정 배음 구조를 지닌 사운드가 살아 움직이는 효과로 공간을 디자인하였다.

Max에서 Syphon을 통해 Arena로 전송된 영상은 transform을 통해 배치를 다르게 한 뒤 edge detection을 거쳐 세포분열과 같은 영상효과를 입혀 주인공이 불규칙적이고 불안정한 공간에 들어와 있음을 표현하였다 [그림-47].



[그림-47] A' 파트 영상효과에 따른 변화 과정

PRIZONE은 시나리오에 따라 4면이 켜졌다가 꺼짐을 반복하며 주인공의 내면을 시각화한다. 처음 몇 번은 천천히 깜빡이다가 tape music의 특정한 소리에 반응하여 LED를 순차적으로 빠르게 세 바퀴를 회전하여 주인공의 회오리치는 내면을 시각적으로 표현하였다.

무빙(moving)<sup>90)</sup>을 통하여 파도의 형태를 가진 고보<sup>91)</sup>로 무대 전체를 채웠다. 느리게 움직이는 바닥을 통하여 심해 속을 떠도는 이미지를 구현하였다.

90) 콘솔의 컨트롤로 포커싱이 가능한 조명

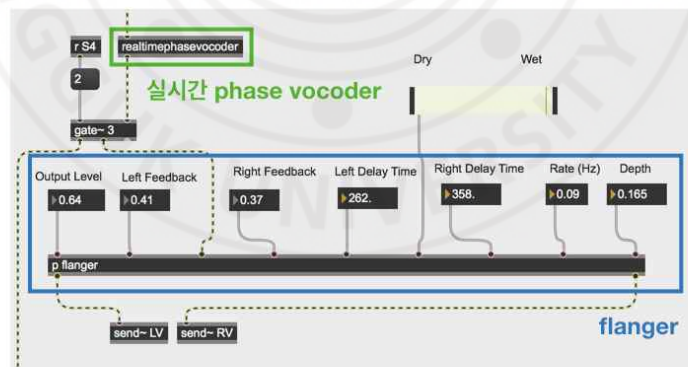
91) 방출 된 빛의 모양을 제어하기 위해 광원 내부 또는 앞에 배치 된 스텐실 또는 템플릿이다.

### 3) B파트

〈표-12〉 B파트 구성

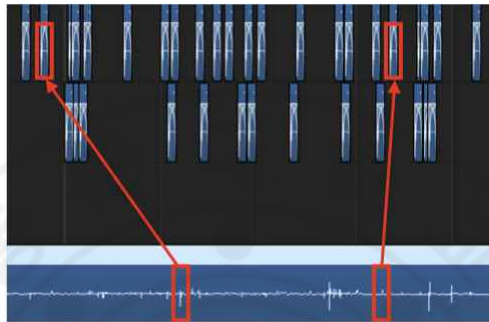
시간	4:48 ~ 5:32
사운드 프로세싱	phase vocoder + flanger
영상효과	edge detection scale blur
PRIZONE 효과	목소리 인터랙션

B파트에서는 실시간 phase vocoder 음향효과를 거친 목소리에 flanger 음향효과를 주었다 [그림-48]. flanger 음향효과로 큰 원을 도는 울림을 통해 관객들에게 이질적인 느낌을 전달하였다. 이를 통해 반전된 사운드를 더욱 강렬하게 만들어 주인공의 열망에 대한 의지를 표현하였다.



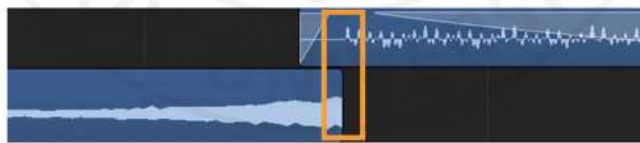
[그림-48] B파트 phase vocoder + flanger 패치

tape music은 [그림-49]와 같이 사운드 샘플에서 원하는 영역을 20ms~40ms로 잘라내어 편집하였다. 물이 바위에 떨어지는 순간의 사운드를 불규칙적으로 편집하여 자잘한 파열음이 연속적으로 나오도록 디자인하였다.



[그림-49] B파트 사운드 소스 편집 과정

[그림-50]은 서로 다른 사운드의 시작과 끝 지점을 연결 및 편집하여 예측이 불허한 사운드를 제작하였다. 고음의 배음이 풍성한 사운드로 전개가 되다가, 저음 배음 구조로 전환되는 과정을 통하여 비현실적이고, 낮은 공간을 청각적으로 연출하였다.



[그림-50] B파트 사운드 소스 연결 과정



영상은 영혼 이미지 속에 영원 이미지의 움직이는 원이 생성된다. 영혼 이미지는 목소리의 음량 값에 의해 인터랙션이 일어나며, 영원 이미지는 tape music의 특정 주파수 음량 값에 의해 변형이 일어난다. 시간이 지남에 따라 영원의 이미지가 점점 더 선명해지며, 에너지가 증폭되고 있음을 연출하고 있다.

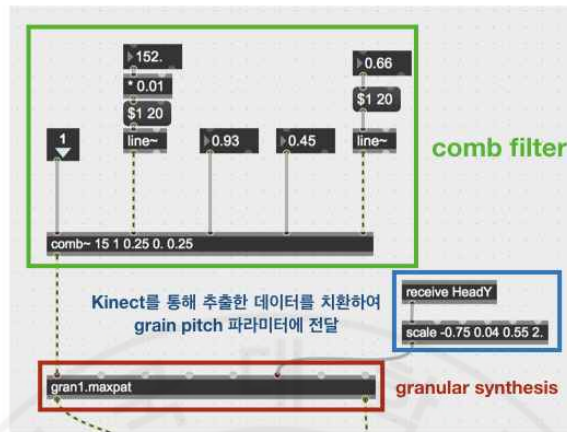
PRIZONE은 목소리의 음량 값에 정비례하여 밝기를 연동시켰다. 주인공의 영원에 대한 의지를 PRIZONE에 투영하여 표현하였다.

#### 4) C파트

<표-13> C파트 구성

시간	5:32 ~ 6:46
사운드 프로세싱	phase vocoder + flanger phase vocoder + comb filter + granular synthesis
영상효과	radial blur goo transform
PRIZONE 효과	움직임 인터랙션

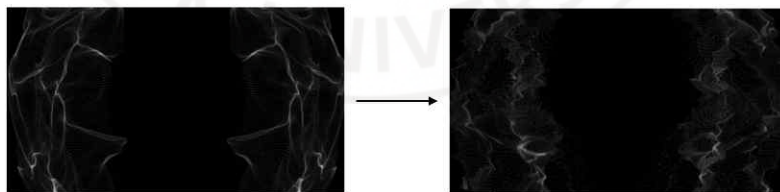
C파트에서 tape music은 점점 더 많은 사운드 층을 쌓기 시작한다. 또한 실시간 phase vocoder에 comb filter와 granular synthesis 사운드 프로세싱을 거친 효과를 추가하여 주었다 [그림-51]. 목소리의 원래 형태는 사라지고 잘게 쪼개어져 영원에 의해 점점 파괴되어가는 과정을 연출하였다. 음원 샘플들의 피치 값은 주인공의 얼굴 높낮이 데이터에 매핑되어, 상승과 하강의 시각적 이미지를 청각화 하였다. 이를 통하여 주인공 행동의 목표에 대한 의미를 보강해주었다. 몸이 관객석으로부터 멀어짐에 따라 재생속도를 한층 더 느리게 반전시킴으로써 삶으로부터 멀어지는 주인공을 나타냈다.



[그림-51] C파트 사운드 및 인터랙션 패치

영원 이미지가 확장되며, 영혼 이미지는 양쪽 사이트의 불길로 나뉘었다. [그림-52]는 Arena의 goo 효과가 적용된 이미지를 나타낸다. goo 효과와 opacity 파라미터 값에 목소리 음량 값을 연동하여 After Effect를 통해 만든 불길이 일렁이고 휘날리며 거세어지도록 제작하였다. 불길은 형벌이며, 자아에 갇혀버린 주인공이 이를 벗어나기 위하여 겪는 고통을 형상화하였다.

goo 효과



[그림-52] C파트 영상에 적용된 goo 효과

배우는 C파트에서 PRIZONE 안으로 빨리 들어가게 된다. 모션 트래킹을 통하여 배우의 좌우 데이터와 앞뒤 데이터를 추출하여 PRIZONE의 LED에 연동하였다. [그림-53]과 같이 LED는 배우의 위치를 계산하여 나가려는 방향을 차단한다. 배우는 어떠한 방향으로도 PRIZONE을 벗어날 수가 없게 된다.



[그림-53] 움직임 데이터를 통한 PRIZONE 연동

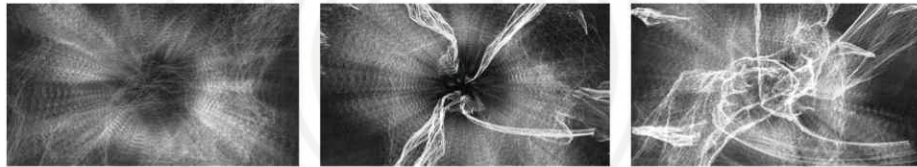
#### 4) D파트

<표-14> D파트 구성

시간	6:46 ~ 7:58
사운드 프로세싱	X
영상효과	flip suckr twisted radial blur trails blur edge detection transform
PRIZONE 효과	random

D파트는 극 전체의 절정 부분으로, 주인공이 거대한 힘에 의해 육체가 파편으로 흩어져버리는 고통을 받게 된다. 온 힘을 쏟아붓는 움직임에 위하여, 대사를 미리 녹음한 뒤 프로세싱을 거쳐 tape music에 추가하였다. tape music의 로우 임팩트 사운드는 주인공의 심장 소리이며, 고음역대는 찢겨져 나가는 아픔을 표현하였다.

이미지는 미리 제작된 영상을 여러 층으로 겹쳐서 사용하였다. After Effect에서 영상 파라미터에 저음역대 로우 임팩트 사운드 소리의 음량 값을 연동하여 힘줄이 생겨났다가 사라지는 효과를 주었다 [그림-54]. Arena에서 다양한 영상효과를 입혀 혼란스러운 영원 속에서 주인공이 흩어져 휘날리는 연출을 하였다.



로우 임팩트 사운드의 음량 값에 따른 영상 변화

[그림-54] D파트 로우 임팩트 사운드의 음량 값에 따른 영상 인터랙션

PRIZONE은 Max를 통해 0 아니면 255의 값을 Arduino로 랜덤(random)하게 전달하여 4개의 LED가 불규칙적으로 깜빡이게 설정하였다. 주인공의 잘게 부서진 자아를 표현하였으며 D파트가 끝나는 시간에 맞춰 암전하였다.

#### 4) A” 파트

〈표-15〉 A” 파트 구성

시간	7:58 ~ 9:50
사운드 프로세싱	phase vocoder + comb filter + granular synthesis
영상효과	scale opacity
PRIZONE 효과	시나리오

A” 파트에서는 실시간 phase vocoder를 거친 comb filter, 그리고 granular synthesis 사운드 프로세싱 효과를 주었다. 주인공이 블랙홀에 빠져 영원히 잠식되는 과정을 그리기 위하여, phase vocoder의 음량 값을 서서히 fade out<sup>92)</sup> 시켜줬으며 granular synthesis 음향효과의 피치 값이 하강하도록 제어하였다.

영상은 Arena와 Max를 통해 미리 제작된 영혼 이미지의 불투명도에 목소리 음량 값을 연동하였으며, 크기 값을 서서히 줄여 영혼의 불이 꺼져가는 연출을 하였다.

PRIZONE은 컨트롤러를 사용하여 밝기 값을 조절해주었다. 컴퓨터 두 대의 네트워크를 공유기에 연결한 뒤, 오퍼레이터 컴퓨터의 Max에서 udpSend 오브젝트로 무대 쪽 컴퓨터에 데이터 값을 전송하였다. 전송된 데이터는 Arduino를 통해 LED의 밝기 값을 조절하여주었다.

92) 음원의 음량 값이 차츰 작아지며 사라지는 효과

#### 4. 기술 적용의 예술적 효과

사운드 디자인을 통해 만든 tape music은 관객들에게 낯설고, 음침한 청각적인 배경을 제시함과 동시에 극 전체를 이끌어갔다. 조성음악<sup>93)</sup>에 익숙한 대중들에게 컴퓨터로 합성한 사운드로 편집된 무조음악<sup>94)</sup>은 관객들에게 새로운 청각적 경험을 선사한다. 전반적으로 이질적이고 낯선 사운드로 제작한 tape music은 배우가 처한 비현실적인 상황과 스토리에 따라 관객이 공감할 수 있도록 도와주었다.

Max를 통해 사운드 프로세싱을 거친 배우의 목소리는 기존에 존재하지 않던 새로운 배음을 생성하여 감정적인 표현의 범위를 확장할 수 있었다. 목소리는 pitch shift 음향효과를 통해 낮은 배음 구조의 음색으로 변형되어, 주인공을 초월적인 공간에 들어와 있는 듯한 연출을 하였다. 실시간 phase vocoder 음향효과를 통해 느리게 반전된 사운드는 어절이 맞물려 출력되면서 대사가 연장되는 효과를 낼 수 있었다. delay와 chorus 음향효과를 이용하여 하나의 목소리가 사방에서 울려 퍼져 나오는 초월적인 경계면에 대한 연출을 하였다. 또한, flanger와 comb filter 음향효과를 이용하여 기존의 음색을 왜곡시켰다. 이를 통하여 자연에 존재하지 않는 사운드를 디자인하였다. 움직임 데이터로 피치 값과 재생속도를 변화 시켜 행위에 의한 시각적인 표현을 청각적으로 도왔다. 배우가 사운드 인터랙션을 위한 연기를 하지 않아도, 자연스럽게 인터랙션이 이루어지도록 시스템을 설계하였다. 사운드 프로세싱을 통해 목소리의 표현에 대한 한계를 보완할 수 있었으며, 극 전체의 연출을 다채로운 방식으로 도울 수 있었다.

---

93) 음악에 쓰이는 화성이나 멜로디가 하나의 음 또는 하나의 화음을 중심으로 하여 일정한 음악관계를 가지고 있을 경우를 말한다.

94) 현대 음악에서 발달한 음악 양식으로 정해진 조성이 없이 연주되는 곡의 형태이다.

사운드와 움직임은 영상과 연동되어 극의 기승전결에 대한 이미지를 구현하였다. tape music은 영원에 대한 이미지에 연동되었으며, 배우의 목소리는 영혼에 대한 이미지에 연동되었다. 이러한 인터랙션을 통해 영원과 영혼 간의 간극을 표현하였으며, 추상적인 개념을 이미지로 구체화해 극의 전체적인 스토리와 배경을 설정하였다. 즉, 관객에게 청각적인 요소와 시각적인 요소가 상호작용하는 뉴미디어를 제공하는 멀티미디어 작품을 제작할 수 있었다.

PRIZONE은 사운드와 배우의 행위에 따라서 LED의 밝기가 변화하였다. 주인공을 가두기 위해 LED가 켜지거나 고통의 절정에서 밝기 값이 랜덤하게 변화하는 등, 각 파트에 따라 다른 프리셋을 지정하여 주인공의 내면이 투영된 인터페이스를 통한 시각적인 연출을 하였다.

## IV. 결 론

본 논문은 실시간으로 프로세싱된 사운드와 그에 반응하는 영상 및 인터페이스를 이용한 인터랙티브 멀티미디어 퍼포먼스 제작에 대한 연구이다. 배우가 지닌 본연의 음색만이 아닌 연출에 따라 다양한 음색을 디자인하였으며, 사운드와 움직임에 연동되는 영상과 인터페이스를 구현하여 뉴미디어적인 요소와 연극적인 요소가 융합된 멀티미디어 퍼포먼스를 제작하였다. 배우의 연기에 따른 인터랙션을 통하여 연극의 전개를 직관적으로 전달할 수 있었으며, 표현의 확장을 통해 관객과 예술적 교감을 효과적으로 나눌 수 있도록 하였다.

Max를 사용하여 tape music을 제작하였으며 실시간 사운드 프로세싱에 대한 연구를 하였다. additive synthesis, frequency modulation, comb filter 음향효과를 이용하여 tape music을 제작하였으며, 어쿠스틱 악기를 통해서 나올 수 없는 사운드를 음악에 적용하였다. 또한 배우의 음성에 stereo delay, flanger, chorus, granular synthesis, pitch shift, phase vocoder 음향효과를 작품의 초목표에 맞춰 새로운 사운드를 디자인하였다. 그로 인해, 배우가 아날로그적으로 표현할 수 있는 연기를 확장하여 고전 연극이 지니고 있던 한계점을 극복할 수 있었다.

영상은 tape music과 실시간 프로세싱된 사운드 데이터를 After Effect, Jitter, Arena에 적용하여 인터랙션을 주었다. 영상 이미지는 각 사운드 요소의 음량 값에 반응하도록 만들었으며, 극의 전개에 따라 변화하였다. 효과적인 영상 표현을 위하여 다양한 영상효과를 사용하였으며, 제작된 영상이 사운드와 인터랙션 되고 있음을 관객이 알 수 있게 하였다. 이를 통해 단순히 극적인 배경을 제시하는 것이 아닌, 주인공의 초목표, 갈등의 요소, 극의 전개와 같은 연극적인 요소와 융합한 미디어를 제작하였다.



연극적 오브제로 제작된 인터페이스는 사운드와 움직임 데이터에 연동하였다. 데이터는 OSC 통신과 Serial 통신을 이용하여 안정적으로 제어하였다. 백색 LED로 주인공의 내면 및 연극적 배경을 구현하여 매체의 특성을 재해석하였다. 그러나 이번 연구를 통하여 발생한 문제점 및 남은 연구과제는 다음과 같다.

첫째는 극장에 영상과 조명, 그리고 인터페이스 간의 밸런스에 대한 문제이다. 영상을 출력하는 빔 프로젝터의 밝기가 조명과 인터페이스에 영향을 주었다. 그로 인해, 기존에 디자인했던 조명을 최소화하고 인터페이스의 밝기도 최대한 높여주었지만, 원하는 만큼의 결과물을 얻지 못하였다. 이후 작품에서는 각 미디어 간의 관계에 대한 다방면의 고려가 이루어져야 할 것이다.

둘째는 움직임 데이터의 안정화 및 범위에 대한 문제이다. Kinect는 적외선을 이용하여 데이터를 추출하는데, 극장의 조명에도 적외선이 존재하여 조도를 높이면 데이터 값이 불안정해지는 결과가 생겼다. 또한 Kinect의 범위가 넓지 못하여, 배우가 인터페이스 내에서 연기할 때에만 데이터를 추출할 수 있었다. 그로 극장 전체의 배우 움직임을 안정적으로 감지하는 시스템을 구축하는 것에 대한 연구가 이루어져야 한다.

Keyword(검색어)

인터랙티브 멀티미디어 퍼포먼스(interactive multimedia performance), Max/MSP, 컴퓨터음악(computer music), 소리시각화(sound visualization), 뉴미디어(new media), OSC(open sound control), 아두이노(Arduino), 피지컬 컴퓨팅(physical computing), LED, 모션 트래킹(motion tracking), 실시간 사운드 프로세싱(real-time sound processing), 키넥트(Kinect)

E-mail: charles92@naver.com

## 참 고 문 헌

### 1. 단행본

- 김영민, 「사운드 디자인을 위한 맥스」, (Real Lies Media, 2017)
- 이석원, 「음악음향학」, (심설당, 2003)
- Michael Barr, 「Introduction to Pulse Width Modulation(PWM)」, (O'Reilly Network, 2003)
- J.M. Chowning. 「The Synthesis of Complex Audio Spectra by Means of Frequency Modulation」, (J. Audio Eng. Soc. 1973)
- Charles Dodge, Thomas A. Jerse, 「Computer Music: synthesis, composition and performance, Second Edition」, (Schirmer Books, 1997)
- Bob Katz, 「Mastering Audio the art and the science , third edition」, (Focal Press, 2007)
- Curtis Roads, 「The Computer for music」, (MIT Press, 1996)

- Randi J. Rost, Bill Licea-Kane, 「OpenGL shading Language third Edition」, (Addison Wesley, 2010)

## 2. 참고논문

- 강현우, 「인도음악 연구를 통한 인터랙티브 멀티미디어음악 제작 연구」 (동국대학교 영상대학원 멀티미디어학과, 2017)
- 라지웅, 「Max/MSP와 Generative Art를 이용한 멀티미디어음악 제작 연구」 (동국대학교 영상대학원 멀티미디어학과, 2018)
- 이보강, 「피아노의 실시간 사운드 프로세싱을 이용한 인터랙티브 멀티미디어 퍼포먼스 연구」 (동국대학교 영상대학원 멀티미디어학과, 2018)
- 조환희, 「베이스 트롬본과 피아노의 실시간 사운드 프로세싱을 이용한 멀티미디어 제작 연구」 (동국대학교 영상대학원 멀티미디어학과, 2019)
- 최준환, 「무용 동작인식을 이용한 인터랙티브 멀티미디어작품 제작연구」 (동국대학교 영상대학원 멀티미디어학과, 2014)
- 한승욱, 「피아노의 실시간 프로세싱을 이용한 멀티미디어음악 제작 연구」 (동국대학교 영상대학원 멀티미디어학과, 2018)

### 3. 웹사이트

- Arduino  
<https://www.arduino.cc>
- CNMAT: external Max object, OSC  
<http://cnmat.berkeley.edu/>
- Max  
<https://cycling74.com/>
- NI mate  
<https://ni-mate.com/>
- Syphon  
<http://syphon.v002.info/>
- Proforma  
<http://www.proformavideodesign.com/news/kinect-performance/>

## ABSTRACT

### A Study on Interactive Multimedia Performance using Max/MSP and New Media Art (focus on Multimedia Music <Eternity>)

Lim, Jun Hyoung

Department of Multimedia  
Graduate School of Digital Image and Contents  
Dongguk University

The purpose of this study is to create theatrical performance converging with new media art by overcoming the time and space limitations and trying to expand the range of expression of the classical play. Each media is connected as one through interaction and leads the play. An actor's words extended the range of expression through sound processing. Computer graphics are linked to the sounds and change their shapes with the volume of sounds. The LED interface interacts with the actor through physical computing. This thesis is divided into four parts; sound design system, real-time video control system,

interface system, and interaction system establishment.

The sound processing used in the study is implemented through Max program. Sound design is divided into two parts. Tape music is produced by editing sound sources which are created with additive synthesis, frequency modulations, etc. through Logic Pro X. Real-time sound processing of the actor's words are created with delay, flanger, chorus, granular synthesis.

Generative computer graphics were made with After effect and Jitter, and were finally printed through Arena. OSC is used to change the shape of graphics. The role of OSC is crucial to communicate with Max and Arena. The shape of graphics is changed with a volume of actor's voice. To do so, OSC is used. Also, for sharing video images between softwares, Syphon is used. Syphon is a communication protocol that allows video frames to be shared on a network.

The interface is manufactured with Arduino and LEDs. The brightness values of the four LEDs are changed depending on the actor's performance. The actor's motion data is extracted via Kinect sensor, which is transferred to Max by OSC. The data is sent from Max to Arduino via Serial Communications for interaction.

Various media allows theatrical production to be expanded and interacts with theatrical elements. It also gives fun through interesting elements to play genres that can be boring. Through this, it is able to produce a multimedia performance that effectively converged with new media technology.

## 부록-1 : Rimbaud <L'Éternité>

### L'Éternité

Elle est retrouvée.  
Quoi? – L'Éternité.  
C'est la mer allée  
Avec le soleil.

Âme sentinelle,  
Murmurons l'aveu  
De la nuit si nulle  
Et du jour en feu.

Des humains suffrages,  
Des communs élans  
Là tu te dégages  
Et voles selon.

Puisque de vous seules,  
Braises de satin,  
Le Devoir s'exhale  
Sans qu'on dise: en fin.

Là pas d'espérance,  
Nul orietur.  
Science avec patience,  
Le supplice est sûr.

Elle est retrouvée.  
Quoi? – l'Éternité.  
C'est la mer allée  
Avec le soleil.

## 부록-2 : 첨부 DVD 설명

1. <Eternity> 공연 영상 : 2019년 11월 16일 이해랑 예술극장 공연 영상
2. <Eternity> Max patch : 작품에 사용된 Max 패치

