석 사 학 위 논 문

# 하모니카 연주의 실시간 사운드 프로세싱과 LED를 이용한 멀티미디어 작품 제작 연구

(멀티미디어음악 작품<Sail>을 중심으로)

지도교수 김 준

동국대학교 영상대학원

멀티미디어학과 컴퓨터음악전공

오 아 영

2 0 2 0

석 사 학 위 논 문

# 하모니카 연주의 실시간 사운드 프로세싱과 LED를 이용한 멀티미디어 작품 제작 연구

(멀티미디어음악 작품<Sail>을 중심으로)

오 아 영

지도교수 김 준

이 논문을 석사학위 논문으로 제출함

2019년 12 월

오아영의 음악석사(컴퓨터음악)학위 논문을 인준함

2020년 1 월

위원장 박 상 훈

위 원 정 진 헌

위 원 김 준

동국대학교 영상대학원

# 목 차

# 표 목 차



# 그 림 목 차

# Ⅰ. 서 론

## 1. 연구의 배경과 목적

미디어(media)는 대중에게 정보를 전달하는 수단을 뜻하며 미디어아트(media art)는 현대에서 사용되는 여러 미디어를 통해 미술에 적용하여 작품으로 표현하는 예술이다.[1] 현재는 규모가 큰 설치형 작품들을 활용하여 미술관, 대규모 페스티벌과 같은 곳에 미디어아트가 도입되어 있으며 대중들이 쉽게 접할 수 있는 예술의 형태가 되었다. 음악, 무용, 연극과 같은 고전 형태의 예술들이 프로젝션 매핑(projection mapping)[2], 미디어 파사드(media facade)[3] 등 새로운 기술과 결합되어 현재는 디지털 인터페이스를 통한 물리적인 상호작용이 가능하기 때문에 다방면의 감각을 활용하여 관객과의 소통을 극대화하는 방향으로 발전되고 있다.

사람은 시각을 통해 다양한 외부 정보를 인지하기 때문에 외부 활동과의 상호작용에서 시각은 중요한 역할을 한다. 따라서 대부분의 미디어아트 작품들은 시각적 자극을 제공하는 방식으로 창작되며, 본 멀티미디어 작품 <Sail>은 음악과 연동된 시각적 설치작품(installation artwork)[4]을 통해 작품화된다. 실시간으로 프로세싱된 소리와 LED[5] 빛이 서로 연동되는

---

1) 최연희, 「현대의 예술과 미학」(서울대학교출판부, 2007) p.35
2) 대상물의 표면에 빛으로 이루어진 영상을 투사하여 변화를 줌으로써 다른 성격을 가진 것처럼 보이도록 하는 기술이다.
3) 건물 외벽에 LED등의 디스플레이를 부착하여 영상을 구현하는 방식.
4) 어떤 소재를 주위 공간과 어울리게 배치하거나 설치하여 만든 예술 작품을 의미한다.

작품으로, 변형된 악기 소리 및 LED 빛의 색과 움직임을 멀티미디어 작품으로 만들었으며 이러한 방식을 통해 빛의 청각을 표현한다. 따라서 LED 빛을 통한 시·청각적 감각이 극대화되어 작품의 전달력이 향상된다.

## 2. 선행 연구

LED는 일반 조명보다 발광이 세고, 발열이 덜하기 때문에 야외공간과 공연장에서 많은 사용을 하고 있다. 다음의 [그림 - 1]은 동대문 디자인 플라자(DDP) LED 정원 사진이다. 2만 송이의 장미는 LED 조명으로 표현하여 정원을 조성했으며, 위의 LED의 장점들이 부각 되어 작품화된 것을 볼 수 있다.

**[그림-1] 동대문 디자인 플라자(DDP) 장미정원**

또한 LED는 기능적으로 유연하게 제작되어 무대에서 화려한 퍼포먼스를 표현할 수 있다. [그림-2]는 America's Got Talent 2017에

5) Light Emitting diode의 줄임말로 순방향으로 전류가 흐르면 빛을 내는 반도체 소자이며 다양한 색깔의 빛을 내고, 다른 조명에 비해 밝다.

보인 <라이트 밸런스> (Light Balance)팀의 공연이다. 몸에 LED를 장착하여 춤과 음악에 맞춰 제어되는 작품이며 몸의 움직임이 많은 퍼포먼스이기 때문에 유연한 재질의 LED 결합은 작품에 화려한 시각적 자극을 준다. 장착된 LED뿐만 아니라 조명, 영상 등을 융합하여 무대를 채우고 있다.

[그림-2] LED를 이용한 Light Balance의 Performance

본 논문의 선행 작품으로는 2019년 6월 <SEEING SOUND LISTENING IMAGE-MEDIA ART EXHIBITION> 전시를 위해 제작된 '구름 운(運)'이 있다. 작품은 아두이노(arduino)[6]와 초음파 거리 센서를 통해 작품과 관람객 사이의 거리에 따라 영상과 음악이 바뀌는 시스템으로 구성되었다. 그러나 초음파 거리 센서의 경우 스피커의 파장(wave)[7]이나 다른 센서의 파장과 겹치게 될 때, 아두이노 값이 불안정하여 제어가 어려운 것을 확인할 수 있었다. 또한 구름의 형태가 종이로 제작되었기 때문에 고정하기 어렵고 외부의 충격에 약하다는 한계점이 있다.[그림-3]

---

6) 물리적인 것을 감지하고 제어할 수 있는 객체들과 디지털 장치를 만들기 위한 도구로, 오픈 소스를 통해 마이크로컨트롤러를 쉽게 작동시킬 수 있다.
7) 주어진 시각에 반복되는 모양을 주기적으로 보이는 파동을 말한다.

[그림-3] 전시작품 '구름 운(運)'

이번 연구에서는 선행 연구 때의 문제점을 보완하고 사운드와 연동되는 인터랙션(interaction) 시스템을 추가하여 무대공연으로 표현하고자 한다. 구조물의 색감과 빛의 움직임을 더 강렬하게 표현하기 위해 프로젝터 영상 대신 LED를 사용하였고, 모형의 내구성을 높이기 위해 소재를 한지 전등과 솜으로 변경하였다.

작품 <Sail>은 악기 연주로 입력되는 사운드 데이터를 활용하여 LED 빛으로 표현하고 소리시각화를 구현한다. 사각형 디스플레이(display)의 영상을 이용한 일반적 공연형태의 틀을 벗어나 구름 모양의 구조물을 제작하여 작품의 주제를 더욱 시각적으로 강화하여 표현할 수 있었다. 본 연구는 선행 연구를 기반으로 LED가 부착된 구조물을 제작하고 음악과 연동된 시스템을 통해 빛의 움직임을 직관적으로 보여준다. 따라서 작품 <Sail>을 통해 음악이 구름의 빛을 이끌어가는 것처럼 표현하고자 한다.

# Ⅱ. 작품 제작 기술 연구

작품 <Sail>에 쓰인 악기는 하모니카의 다양한 종류 중, 음정을 뚜렷하게 연주할 수 있는 크로매틱 하모니카(chromatic harmonica)[8]를 사용했다. 작품에서 전반적으로 사용된 주법은 다음과 같다.

**<표-1> 하모니카의 주법과 설명**

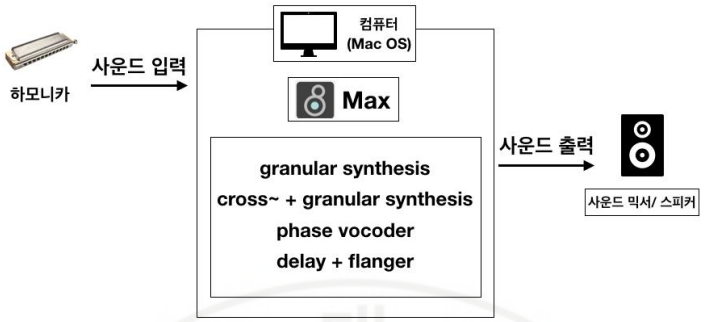
| 주법 | 설명 |
|---|---|
| 텅블럭 (tongue blocking) | 입술이 세 개의 구멍을 덮고, 소리 나는 구멍 하나를 제외한 나머지 구멍을 혀로 막는 방법 |
| 파커 (puckering) | 한 번에 하나의 구멍에 숨을 부는 방법 |
| 텅잉 (tonguing) | 혀를 사용하여 관악기를 연주하는 방법 |
| 글리산도 (glissando) | 두 개의 음을 연결해주는 기법 |

## 1. 사운드 프로세싱 연구

하모니카의 실시간 사운드 프로세싱 음향효과들은 Max[9]를 통해 제작하였다. 아래의 [그림-4]는 실시간 사운드 프로세싱을 위한 시스템이다.

---

8) 반음과 같은 음의 표현이 가능한 악기이다.
9) Cycling'74에서 개발 하였고, 음악과 멀티미디어를 위한 환경을 제공하는 소프트웨어이다. 데이터의 프로그래밍이 가능한 Max, 음향신호 및 MIDI데이터를 처리하는 MSP로 이루어져 있다. 본 연구에서는 최신 버전인 Max8을 사용한다.




[그림-4] 실시간 프로세싱 시스템

작품에서는 한 대의 악기로 연주하지만 여러 대의 하모니카 연주를 표현하기 위해 대표적으로 granular synthesis[10] 방식을 사용했고, 앞서 언급한 주법들 특성에 맞춰 다양한 음향효과를 만들었다. 텅잉 주법으로 화음을 연주하기 위해 granular synthesis를 사용하여 두 음 이상의 소리를 내지만 텅잉 주법의 특성상 입력되는 음량 값이 강하여 고음역대를 감쇄시키기 위해 cross~오브젝트[11]를 이용했다. 또한 글리산도 주법은 phase vocoder[12]의 실시간 reverse[13]를 사용하여 음계를 반대로 연주하는듯한 효과를 적용했으며, 부가적인 사운드 요소로 delay[14]와 flanger[15]를 통해 바람 소리와 같은 사운드 효과를 실시간으로 제작했다.

10) 입력된 사운드를 아주 작은 단위 샘플로 나누어 재조합하여 새로은 사운드를 제작하는 소리 합성방식이다.
11) 원하는 주파수를 기준으로 저음과 고음을 나눠 사운드 데이터를 출력하는 Max 오브젝트이다.
12) 입력된 사운드를 분석하고 데이터를 재합성하여 음의 길이, 높낮이를 변형시키는 소리 합성방식이다.
13) 재생 방향을 역방향으로 만들어 주는 음향효과이다.
14) 입력된 사운드를 지정한 시간 후에 출력하는 효과를 말한다.
15) 딜레이를 활용한 음향효과이고, 정상적인 신호와 딜레이의 시간차를 이용해 새로운 신호를 혼합하여 만든 효과를 말한다.

### 1) 하모니카의 싱글 및 텅잉 주법을 위한 granular synthesis

granular synthesis는 입력된 신호를 작은 샘플 단위로 나누어 재조합하는 소리 합성 방식이다. granular synthesis는 Max의 munger~오브젝트[16]를 사용하여 구현할 수 있다. munger~오브젝트는 7개의 파라미터를 사용하여 수치를 조절할 수 있으며, 작품에서는 주로 샘플 조각의 간격(grain separation), 샘플 조각의 크기(grain size), 샘플 조각의 음정(grain pitch)과 변화(variation)의 값을 조절하여 입력된 사운드를 실시간으로 변형했다. 언급된 파라미터 중, 샘플 조각의 음정 변형은 2의 제곱근의 지수에 따라 계산되어 적용되며 옥타브 간격으로 음정 조절이 가능하다. 아래 <표-2>는 grain pitch 파라미터의 설정 단위로, 음역 구분은 다음과 같다.

**<표-2> grain pitch 파라미터값에 따른 음역**

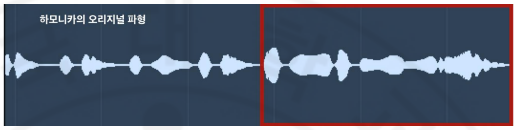
| 적용 파라미터값 | 음역 |
|---|---|
| -n | n 옥타브 아래 |
| 0.25 | 두 옥타브 아래 |
| 0.5 | 한 옥타브 아래 |
| 2 | 한 옥타브 위 |
| n | n 옥타브 위 |

하모니카의 싱글 주법인 텅블럭과 파커를 오리지널 사운드보다 긴 음향으로 변형하기 위해, 위의 granular synthesis의 파라미터를 조절했다.

16) Granular 합성방식을 구현할 수 있는 Max의 외부 오브젝트이며, 내장된 오브젝트가 아니기 때문에 Max의 package manager에서 다운받아야 한다.

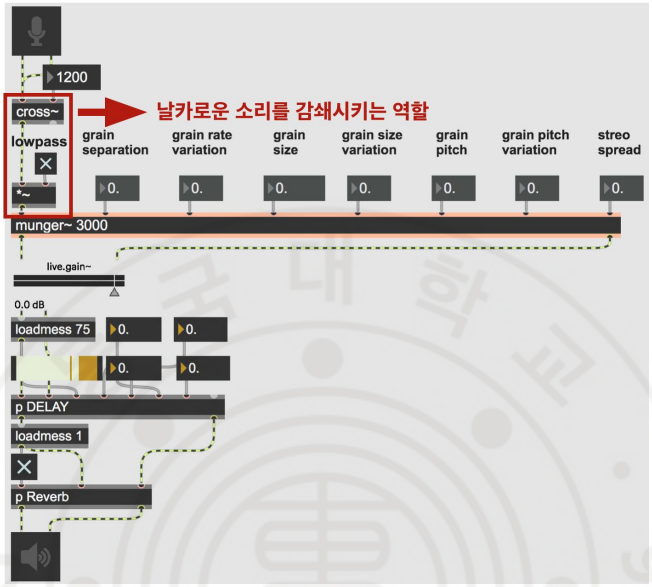
다음은 하모니카의 오리지널 파형과 munger~를 통해 변형된 텅블럭과 파커 주법을 비교한 표이다. 오리지널 파형에서 볼 수 있는 하모니카의 짧은 음가의 엔빌로프(envelope)[17]가 munger~를 거치면 길게 변한 것을 볼 수 있다.

**<표-3> granular synthesis에 따른 파형 비교**

| | |
|---|---|
| **오리지널 파형** |   |
| **munger~로 변형된 파형** |   |

하모니카의 텅잉 주법을 사용하여 화음을 연주할 때, 텅잉 주법만의 짧고 강한 음 처리와 효과적인 리듬을 표현하고자 했다. 하지만 화음과 주법을 동시에 사용하여 munger~로 입력했을 때, 날카롭고 강한 고음역 부분으로 인해 불안정한 사운드가 출력되었다. 따라서 munger~로 소리를 보내기 이전에, cross~오브젝트를 거쳐 1200Hz의 주파수보다 낮은 소리를 출력하고 munger~와 합성되도록 했다. [그림-5]는 cross~에 granular synthesis를 연결한 Max 패치이다. 1200Hz의 저음 부분을 이용하여 하모니카의 텅잉 주법과 어울리는 사운드를 munger~의 파라미터 수치를 이용해 만들었다.

17) 파형의 끝을 서로 연결하여 파형을 둘러싸듯이 그려진 선이고, 사람이 악기 종류를 구분해서 들을 수 있는 중요한 요소이다.



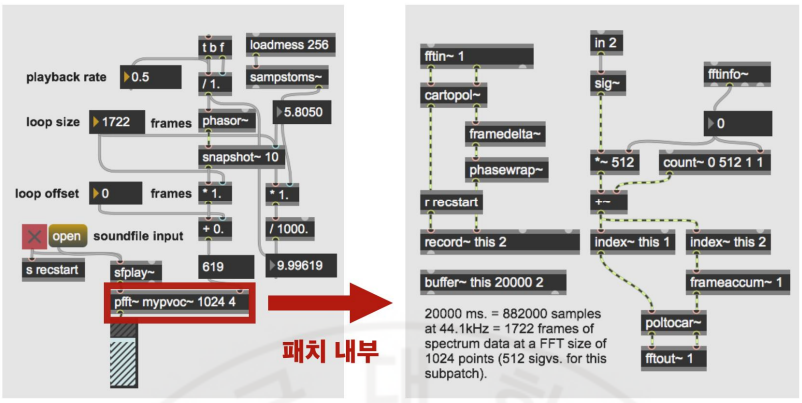

[그림-5] granular synthesis의 패치

## 2) 하모니카의 글리산도 주법을 위한 phase vocoder

phase vocoder는 입력된 사운드를 FFT[18] 분석하여 재조합하고, 음의 높낮이 변화를 주지 않고 재생속도를 조절해주는 소리 합성법이다.

아래의 [그림-6]은 phase vocoder의 기본 패치와 `pfft~mypvoc~`의 패치 내부의 사진이다.

18) Fast Fourier Transform의 약자이며, 고속 푸리에 변환이다. 시간 영역의 사운드 데이터를 주파수 영역으로 옮겨서 구성하고 있는 주파수를 분석할 때 사용 한다.



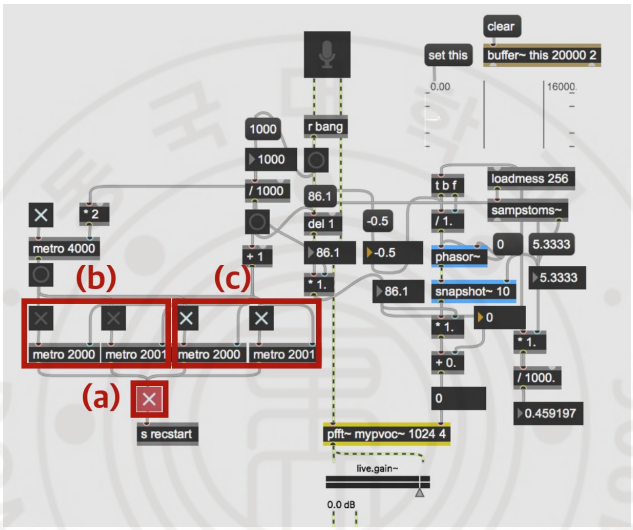

[그림-6] phase vocoder 패치와 pfft~mypvoc~의 패치 내부

사운드가 입력되면 오른쪽 사진의 패치 내부를 통해 FFT 분석이 되고 재생속도(playback rate)와 루프 사이즈(loop size)를 지정해 줄 수 있다. pfft~mypvoc~에서 원하는 음높이를 유지하며 속도를 내보내기 위해서 재생속도에 원하는 속도의 값을 입력해야 하며, 패치 내부의 index~오브젝트[19]를 통해 샘플 단위로 원하는 속도를 읽을 수 있다. 예를 들어 재생속도를 1로 입력 하면, phasor~오브젝트[20]에서 0~1까지의 데이터가 1초 후에 index~에 전송되고, 0.5를 입력하면 2초 후에 데이터를 전송시킨다. 이때 FFT 분석은 프레임(frame) 단위를 사용하기 때문에 루프 사이즈의 단위도 프레임으로 지정해 주어야 한다. 44100 sample rate[21]를 사용했을 때 20초는 1722프레임이 되고, 2초로 계산하게 되면 172.2프레임이 나온다.

하모니카의 주법 글리산도와 phase vocoder의 실시간 reverse

19) 샘플의 수를 나타내고, 입력으로 신호를 수신하는 오브젝트
20) sawtooth의 파형을 생성하는 오브젝트. 0과 1의 사이를 반복하며, 지속적인 반복의 움직임을 결정한다.
21) 1초 동안 추출하는 sampling의 횟수를 의미한다.

음향효과를 만들어 주기 위해 루프 사이즈와 재생속도를 지정해 주었다. 글리 산도 주법으로 음계가 올라가면 reverse가 되어 다시 내려오는 효과의 재생속도는 -0.5의 수치를 입력하고, 1초 단위의 프레임을 사용하기 위해 86.1로 지정해 주었다.




[그림-7] phase vocoder의 reverse를 위한 패치

위의 [그림-7]에서 (a) 부분은 녹음 버튼으로 이 버튼이 꺼지지 않고 지속시키기 위해서는 metro오브젝트[22]를 이용해 주었다. metro의 값을 2000으로 지정해 주어 2초 동안은 녹음 버튼이 켜질 수 있게 지속한 후 2초 동안은 꺼짐을 반복하게 하였다. 실시간 reverse 음향효과를 위해 녹음 버튼이 꺼지지 않게 유지하여 새로운 버퍼를 씌워야 하며, metro를 2000ms와 2001ms의 반복으로 녹음하는 동시에 재생할 수 있도록 (b)의 부분을 만들었다. 하지만 (b) 부분만으로는

22) 입력한 수치에 의해 일정한 간격으로 신호를 내보내는 오브젝트

(a)의 녹음 버튼이 꺼지는 시간이 늘어나게 된다. 따라서 늘어난 시간을 줄이기 위해 (c)의 부분을 추가하여 (a)의 녹음 버튼이 꺼지지 않도록 만들었다. (b) 부분에서 (c) 부분으로 넘어가는 것을 반복하게 하여, 녹음하는 동시에 실시간으로 reverse를 재생할 수 있도록 만들었다.[23]

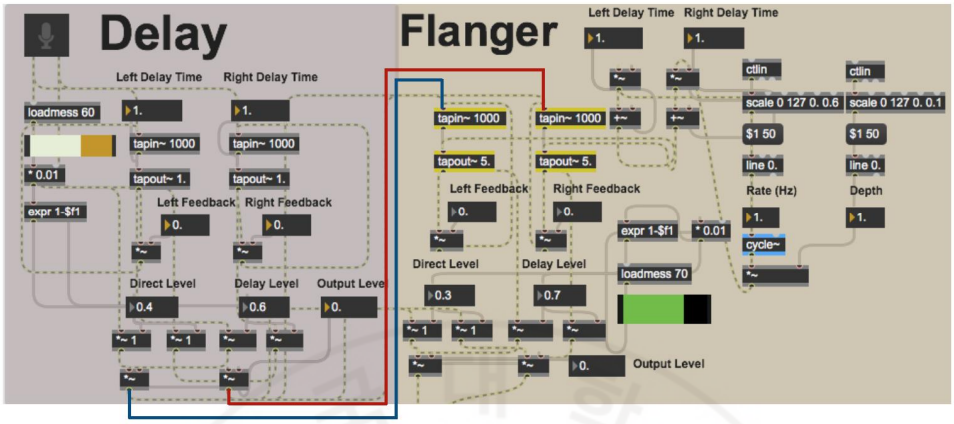
### 3) 하모니카의 음향효과를 주기 위한 flanger

flanger와 delay는 오디오 이펙트(audio effect)[24]의 종류이다. 입력된 사운드의 신호와 짧은 시간의 격차를 이용해 위상차를 만들어 내며, 주파수 대역에 따라 진폭이 증가 되거나 감소 되어 음색을 변화시킨다. 짧은 delay 타임(1ms~10ms)을 이용해 원래의 음색과 합하여 특정 주파수가 강조된 독특한 사운드를 만들 수 있고, delay 타임의 설정에 따라 발생하는 대역이 변하기 때문에 악기 음역에 맞춰 설정해주어야 한다. delay 타임이 길어지면 저음역대, 짧으면 고음역대에서 효과가 생긴다.[25]

아래의 [그림-8]은 작품에 적용된 delay와 flanger가 결합 된 패치이다. delay를 이용하여 하모니카의 오리지널 사운드와 다른 음색을 만들어 공간감을 만든 후, flanger 효과를 같이 결합하여 악기의 소리의 음색을 변화시킨다.

---

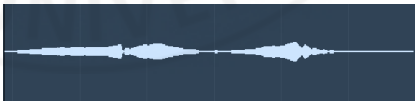
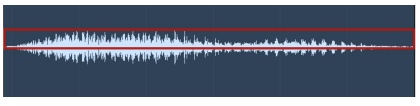
23) 강현우, 「인도음악 연구를 통한 인터랙티브 멀티미디어음악 제작 연구(멀티미디어음악 작품<Drawing Down the Moon>을 중심으로」, 동국대학교 영상대학원: 멀티미디어학과, 석사학위논문, 2017, p.19
24) 특정 오디오의 신호를 변화시켜 음색이나 음질을 바꾸는 것을 말한다.
25) 김근호, 「오디오 용어사전」(새녘출판사, 2013) p.20



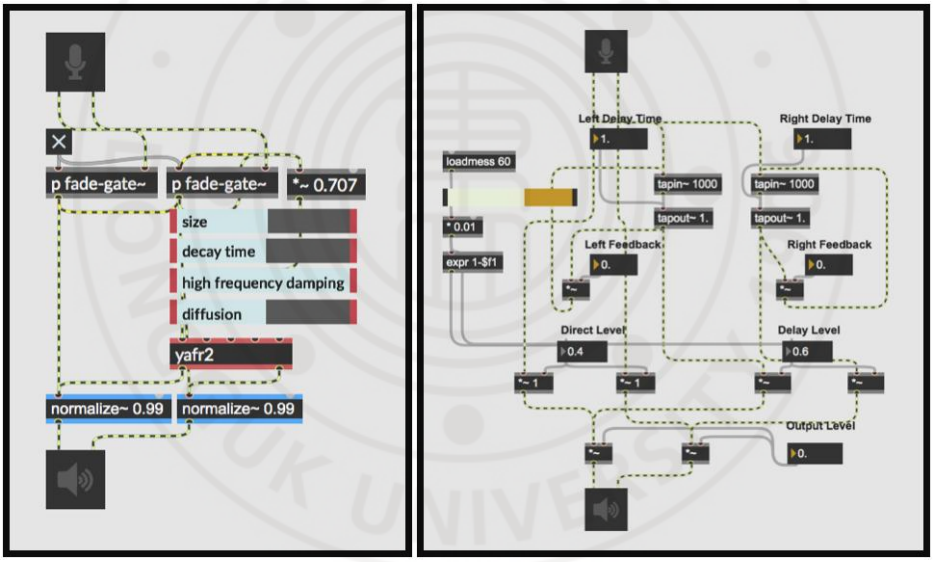

[그림-8] delay와 flanger가 결합 된 패치

delay를 이용하여 공간감을 만든 하모니카의 변형된 사운드와 flanger의 delay 타임을 길게 주어 만들어진 사운드가 합쳐져 위상이 변조된다. 긴 시간의 격차를 이용하여 저음역대에서 변화가 생겨 마치 바람 소리처럼 들리며 하모니카의 음색이 변화되는 효과를 가지게 된다. <표-4>는 하모니카의 오리지널 파형과 사운드 프로세싱을 이용해 바람 소리를 만든 파형의 비교이다.

<표-4> 하모니카 음향효과의 파형 비교

| | |
|---|---|
| 오리지널 사운드 파형 |  |
| 사운드 프로세싱 바람 소리의 파형 |  |

실시간으로 제작되는 바람소리에 공간감을 주기 위해, 부가적으로 reverb와 delay를 사용하였다. 악기의 하모니카 사운드는 sustain이 짧기 때문에 연주 중간에 reverb의 효과를 더해 풍부한 사운드를 더했다. 다음의 [그림-9]는 Max의 reverb와 delay를 구현한 패치이다. 공간의 크기(size), 감쇄시간(decay time), 고주파 대역 소리의 강도(high frequency damping), 소리의 발산(diffusion)으로 조절할 수 있게 구성되었다. reverb는 악기의 오리지널 사운드와 다른 음향효과에 풍부하고 지속적인 효과의 역할을 한다.




[그림-9] rverb와 delay의 패치

delay 패치는 tapin~오브젝트[26]와 tapout~오브젝트[27]를 사용한다. tapin의 값에 사운드 신호를 저장한 후 tapout의 delay

26) 입력되는 오디오 신호를 저장하고 연속적으로 내보내는 오브젝트
27) 지정한 딜레이 시간에 맞춰 저장된 신호의 데이터를 읽는 오브젝트

타임의 값에 따라 재생된다. 왼쪽 delay 타임과 오른쪽 delay 타임에 따라 스테레오 delay 효과를 만들고, 지연된 신호는 다시 tapin~오브젝트로 입력되어 사운드의 피드백 효과를 만들어 낸다. expr오브젝트[28]의 수식을 통해 오리지널 사운드의 원본 사운드와 delay 효과가 거친 변형 된 사운드를 wet/dry로 조절할 수 있다.

28) 입력된 수를 계산식으로 연산해주는 오브젝트

## 2. LED 제작과 연구

네오픽셀(neopixel)[29]의 종류는 스트랜드(strand), 매트릭스(matrix), 링(ring), 스트립(strip), 바(bar) 다양한 종류가 있다. 보통 LED의 경우에는 1개 이상의 단자로 제어가 복잡하지만, 네오픽셀은 하나의 제어용 핀으로 쉽게 제어할 수 있도록 설계가 되어있어, 본 작품 구현을 위해 스트립 네오픽셀을 선택했다.[그림-10]

본 작품에서는 큰 구조물 안에 스트립 타입의 픽셀은 적게는 200개에서 많게는 240개까지 LED를 제어해야 한다. 1m에 30개의 픽셀(pixel)이 있는 스트립 7m와 1m에 60개 픽셀의 4m 스트립을 사용하였다. 4m의 스트립은 픽셀의 사이가 더 촘촘하여 빛의 섬세함을 더욱 살릴 수 있었다. 본 장에서는 LED 연구의 내용을 아래에 자세히 설명하려 한다.

[그림-10] 스트립 네오픽셀

29) 중국의 제조회사 World-semi에서 만든 LED 종류이고, 미국회사 Adafruit는 네오픽셀의 이름과 합작하여 아두이노 라이버리를 개발하고 제공한다.

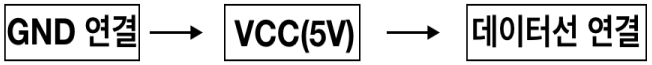
### 1) Neopixel LED 연구

네오픽셀 LED는 아두이노 혹은 아두이노가 호환되는 컨트롤러를 사용해야 한다. 한 픽셀에 RGB의 색상과 LED를 제어하는 칩(chip)[30]이 내장되어있다. 칩은 개발을 통해 전원(VCC)과 그라운드(GND)를 빼고 1개의 데이터 신호만으로 제어가 되고, 많은 픽셀을 개별적으로 조절해주어 다양한 색상의 빛을 만들어 준다. 최근에 나온 WS2812B의 칩은 총 4개의 핀이 있다. <표-5>는 용어에 대한 설명이다.

**<표-5> Neopixel PIN의 용어**

| Neopixel PIN function | |
|---|---|
| VCC(5V) | 전원 공급 |
| DIN | 디지털 인풋 |
| DO | 디지털 아웃풋 |
| GND | 전원 제어 |

이 칩의 경우에는 중간에 LED의 픽셀이 고장이 나더라도 끊기지 않고 동작을 가능하게 해준다. LED는 고장이 잦아 +, -, 그라운드를 뒤집어 연결하게 되면 반대로 흐르는 전류로 인해 사용할 수 없게 된다. 만약 잘못 연결되어있을 경우, 칩이 발열되어 연기가 나는 현상이 나타난다. 따라서 아래의 [그림-11]의 순서로 연결하는 것이 좋다.




**[그림-11] 네오픽셀 연결 순서**

30) 트랜지스터와 같은 반도체가 수십 개로 이루어진 0.5cm 크기의 얇은 조각을 말한다.

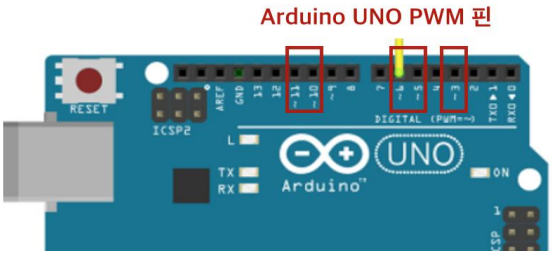
설치와 연구를 할 때 가장 중요하게 여긴 부분은 외부 전원 공급이다. 데이터와 그라운드의 연결이 잘 못 되더라도 픽셀에 지장은 없지만 전원 공급에서 주의해야 하는 것은 작품에 사용되는 전류의 세기 단위(mA)[31]이다. 아두이노의 USB를 통해 전원을 공급받을 때는 500mA의 전류를 사용할 수 있다. 네오픽셀의 LED 1개의 소비량은 20mA이며, 네오픽셀은 LED가 RGB 세 가지로 분류되기 때문에 한 픽셀당 60mA가 필요하다. 아두이노의 전원으로는 8개, 아두이노에 DC잭을 연결할 때에는 16개의 출력이 안정적이다. 그 이상의 픽셀을 사용하게 되면 발열이 발생한다. 작품에서는 5V, 2A의 어댑터를 세 개 사용하여 네오픽셀의 스트립을 전원이 고르게 가도록 하였다.

### 2) Neopixel LED와 Arduino 연결

아두이노의 데이터를 받은 첫 번째 LED 픽셀에서 다음의 픽셀로 하나씩 연결하는 것을 데이지체인(Daisy chain)[32] 방식이라고 하며, LED의 아두이노 제어와 밝기의 제어를 위해 PWM[33]의 방식을 사용해야 한다. [그림-12]의 아두이노 우노[34]에서는 표시된 디지털 숫자가 PWM의 방식으로 지원된다.

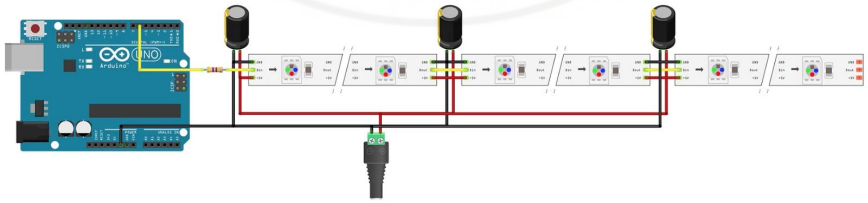
---

31) 전류의 세기 단위이다.
32) 신호를 전송할 때 신호를 요구하지 않는 장치에서 버스를 통해 신호를 전달시키는 방법이다.
33) 펄스 폭의 변조를 말한다.
34) 대중적으로 많이 사용되고 있는 오픈소스 하드웨어로 디지털 기기 발명에 활용된다.




[그림-12] 아두이노 우노의 PWM 핀

네오피셀 스트립 쪽에서는 두 묶음의 선이 있다. 한 묶음은 외부 전원을 연결해주고, 다른 한쪽에서는 아두이노 쪽과 연결하게 되어있다. 스트립의 5V, DIN, GND는 아두이노를 통해 연결할 수 있지만, 외부전압을 따로 사용하기에 5V의 선은 연결하지 않는다. 여기서 아두이노 쪽의 데이터에서 첫 번째 픽셀의 데이터 선과 연결해주는 라인 사이에는 저항이 필요하고, 기본적으로 300~500Ω의 저항을 사용해야 한다. 아두이노에서 나오는 전기 신호가 5V를 넘어 전압이 출력되는 경우 피크(peak)[35]가 발생하여 첫 번째 픽셀이 고장이 나는 경우가 많다. 그것을 막기 위해 저항을 설치하는 것이 안전하다. Max 사운드 데이터와의 인터랙션을 위해 아두이노 코딩을 선택했다. [그림-13]은 아두이노와 여러 개의 네오픽셀 스트립이 연결되어있는 사진이다.

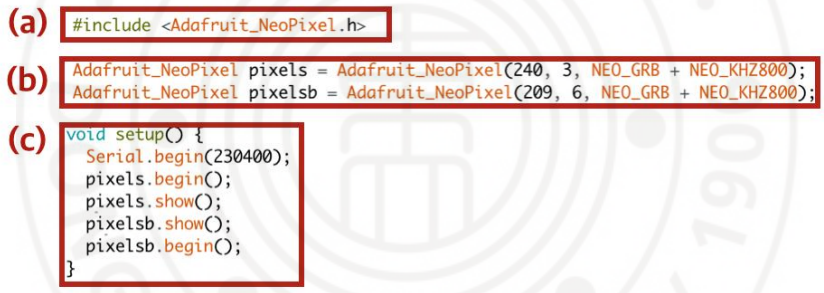


[그림-13] 아두이노와 여러 개의 네오픽셀 스트립 연결

35) 전기나 음향에서 음량이나 전압, 전류의 최대치를 가리키는 경우이다.

### 3) Arduiono와 Max의 제어

네오픽셀은 아두이노 프로그램에서 Adafruit Neopixel의 기본 라이브러리 받아 스트립의 고장 상태 여부를 알아볼 수 있다. 아래의 [그림-14]는 시리얼(serial)을 사용하여 Max의 값을 보내주는 아두이노 코딩이다. 아두이노 상단의 스케치에서 라이브러리 포함하기를 들어가 Adafruit Neopixel을 제공하는 (a)를 부른 후 객체를 부르기 위해 (b)를 입력하고 (c)에서 아두이노와 Max의 시리얼을 동일시시켜 값을 입력받는다.



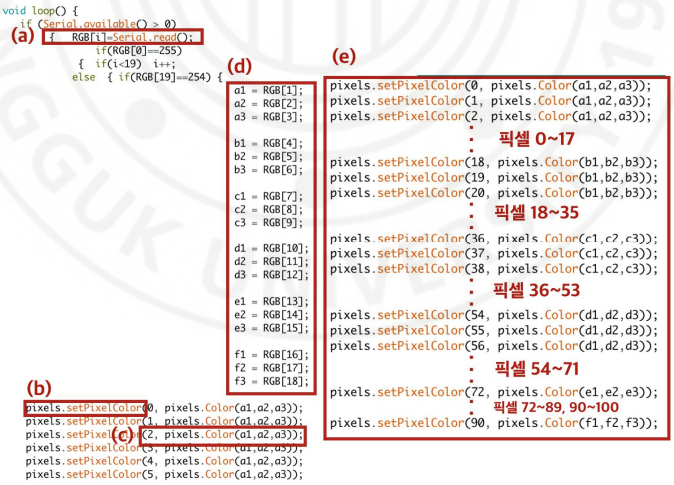

[그림-14] 시리얼을 사용하여 입력 값을 받는 아두이노 코딩

## 3. 공연 시스템 연구

### 1) 사운드와 LED의 연동 방식

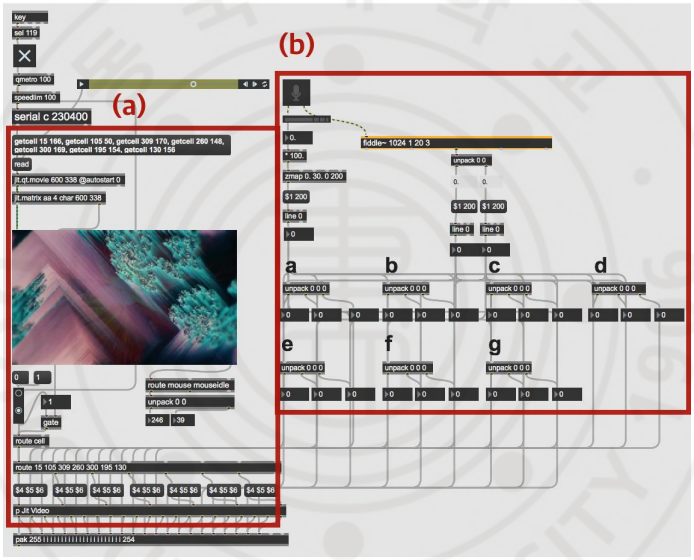
아두이노에 네오픽셀의 영역을 나누어 빛의 밝기를 제어하고, RGB 색을 배합하기 위해 아래의 [그림-15]처럼 코딩을 입력한다.

아두이노에서는 0~255의 값으로 LED의 빛의 밝기를 조정한다. (a) 부분은 시리얼을 통해 Max에서 들어오는 입력값을 읽는다. (b) 부분은 pixels(지정한 스트립의 이름), setPixelColor(픽셀의 컬러를 지정)를 뜻하고 (c) 부분은 픽셀의 번호, 지정된 스트립의 이름, 컬러 값을 각각 R, G, B로 입력한다는 뜻이다. (d)는 (c) 부분에서 픽셀 번호의 RGB 입력을 위해 a1은 Red, a2는 Green, a3는 Blue로 나누어 a부터 f까지 원하는 값을 제어할 수 있는 코딩이다. (d)는 스트립을 제작할 때 원하는 부분을 어떻게 나눌지 미리 생각해 두고 코딩을 만들어야 수정할 때 어려움이 없다. (e) 부분은 네오픽셀 스트립의 원하는 부분의 색감을 어떻게 제어할 수 있는지 확인해 볼 수 있는 코딩으로 예를 들어, 100개 이상의 픽셀을 0 ~17은 a의 RGB, 18~35는 b, 36~53은 c, 54~71은 d, 72~89는 e, 90~100은 f로 나누어 볼 수 있다.




[그림-15] Max와 연동을 위한 아두이노 코딩

아두이노에서 코딩이 완료되면 Max에서 같은 시리얼 번호를 받아 네오픽셀 스트립이 작동된다. 다음의 [그림-16]은 아두이노와 연동되는 Max의 패치이며, 시리얼로 보내는 Max의 값은 두 부분으로 나누어진다. Jitter[36]의 영상을 재생시켜 아두이노로 보내는 (a) 부분과 주파수와 음량 값의 따라 실시간으로 RGB 색감을 만들어 보내주는 (b) 부분으로 나뉜다.

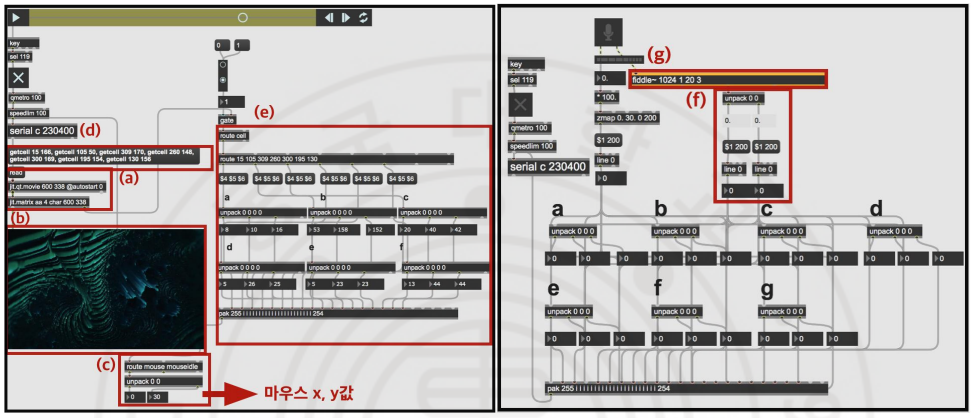



[그림-16] 아두이노와 연동된 Max 패치

Max 패치를 살펴보면 두 부분으로 나누어진다. 우선 첫 번째로 [그림-17]은 LED를 제어하는 Max 패치이다. (a) 부분에서 영상을 재생키면 (b) 부분에서 영상이 나오게 된다. (c) 부분은 영상의 가로 x와

36) Cycling'74에서 개발 하였고, 음악과 멀티미디어를 위한 환경을 제공하는 소프트웨어이고, real-time video와 2D/3D 그래픽을 다루는 Jitter를 사용할 수 있다.

세로 y의 부분을 마우스 움직임을 통해 숫자로 볼 수 있다. 원하는 색감과 움직임의 수치를 (d) 부분처럼 메시지로 불러오고 (e) 부분에서 (d)의 메시지로 들어오는 값을 RGB로 나누어 입력된다. 따라서 영상의 변화로 LED의 색감과 움직임이 출력된다.




[그림-17] LED를 제어하는 Max 패치

(f) 부분에서 들어오는 음정(pitch)값과 음량(amplitude)값을 통해 색을 설정해준다. LED 스트립의 나누어진 a~f부분에 RGB값을 나누어 원하는 색을 지정해 주게 되면 음정과 음량의 값을 통해 실시간으로 빛이 제어된다. 실시간 RGB 배합은 (g) 부분의 fiddle~오브젝트[37]를 통해 하모니카 연주의 실시간 음정과 음량의 값을 알 수 있다. 작품에서 a~f로 나누어진 스트립 부분에 원하는 색을 실시간으로 들어오는 값을 RGB로 분배하여 만들 수 있다. <표-6>은 실제 fiddle~오브젝트로 들어오는 값을 보여주는 표이다.

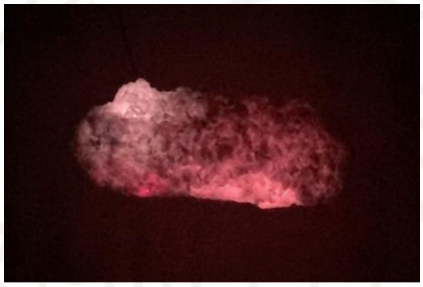
37) 실시간으로 음의 높낮이, 음량, 배음을 분석하는 오브젝트

<표-6> 음정과 음량 값의 RGB 색

| 스트립 | 음정과 음량 값 (RGB) | 색 |
|---|---|---|
| a | pitch-69, amp-70 (69,0,70) | |
| b | pitch-64, amp-80 (80,64,0) | |
| c | pitch-75, amp-90 (0,75,90) | |
| d | pitch-72, amp-95 (72,90,0) | |
| e | pitch-72, amp-100 (0,72,100) | |
| f | pitch-64, amp-100 (64,100,100) | |

Max의 사운드 데이터는 아두이노로 보내지고, 두 프로그램의 연동을 통해 LED의 원하는 부분에 빛의 색이 입력된다. LED의 구성은 section 1~section 4로 나누어진다. 4가지의 배합에 따라 구름의 색과 움직임이 다양해진다. 네오픽셀 스트립의 나누어진 부분에 원하는 색을 입력하거나, 영상제작으로 원하는 색과 움직임을 표현하고, 하모니카 소리의 음정과 음량의 값에 따라 색이 실시간으로 바뀔 수 있도록 하였다. LED의 section 구성을 <표-7>에 정리하였다.

<표-7> LED의 section 구성

| section의 번호 | LED의 이미지 | 표현의 구성 |
|---|---|---|
| section 1 |  | 색감의 배합으로 a~f 부분 중 원하는 곳에 빛이 밝혀지게 하였다. |
| section 2 |  | Jitter의 movie 재생을 이용하여 음악 구성에 어울리는 색감을 영상으로 제작하고 실행하였다. |
| section 3 |  | 하모니카 소리의 음정과 음량 값의 조합으로 인터랙션을 구성하였다. |
| section 4 |  | 원하는 색감의 배합으로 부분을 나누지 않고 구름 전체에 같은 값을 주었다. |

## 2) 구조물 제작과 공연 시스템

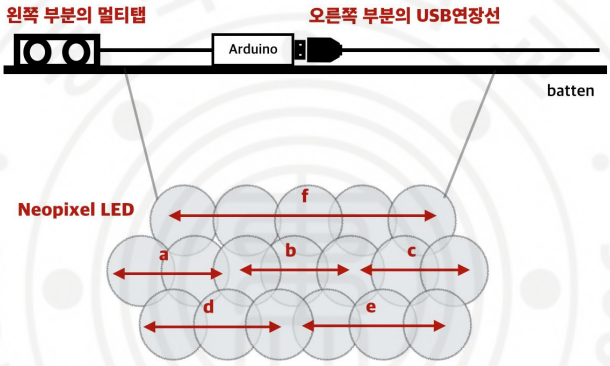
### ① LED 구조물 제작

구조물을 만들기 위해서 지름 30cm의 한지 전등 18구를 사용하였고, 구름을 표현하기 위한 겉의 표면은 솜을 이용하였다. 네오픽셀 스트립의 유연성을 이용해 한지 전등 속에 스프링 형태로 모양을 고정했다. [그림-18]은 구조물 만드는 과정의 사진이다.

**[그림-18] 구조물 제작과정**

구름 구조물을 공중에 매달기 위해서는 조명을 고정할 때 사용하는 바텐(batten)을 먼저 확인하였다. 바텐의 높이, 구름의 설치를 위한 와이어의 길이, 컴퓨터가 아닌 외부전압을 이용한 LED 전원 전기 공급, 아두이노에 입력 값을 주는 USB 연장선의 실험과 길이 등 설치를 위해 여러 방법으로 접근하였다. [그림-19]는 무대에서 생각한 설치 구성도이다. 특히 본 공연의 사운드와 LED의 제어는 객석 뒤에서 이루어지기 때문에 아두이노의 입력 값을 주기 위해 다양한 방법을

구상하였다. 만약 데이터를 보내는 USB 연장선의 길이가 되지 않거나, 값을 받지 못하는 경우 랜(LAN)[38]선을 납땜[39]하는 방법을 사용할 수 있다. 무대에서는 50m의 USB 연장선을 이용하여 오퍼레이터 자리에서 구조물까지 데이터의 값이 튀지 않고, 레이턴시(latency)[40]가 발생하지 않았다. 왼쪽 바텐에서는 전기를 공급하기 위한 멀티탭과 선을 고정 시키고, 오른쪽은 USB 연장선을 고정 시켰다.



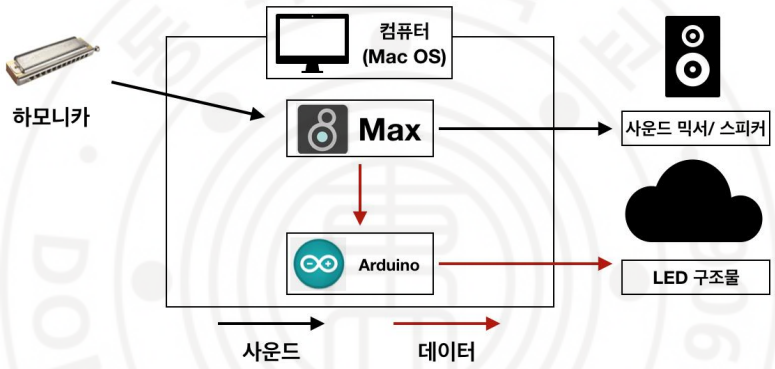

[그림-19] 무대의 구름 설치 모습

### ② 공연 시스템

마이크를 통해 들어오는 하모니카의 사운드는 컴퓨터로 입력된다. 컴퓨터에 입력된 사운드는 Max로 전달 되어 음향효과들이 적용되고, 악기의 오리지널 사운드와 프로세싱 사운드가 합쳐져서 믹서를 통해 스피커로 출력한다. LED의 제어는 입력된 사운드 신호를 Max에서

38) 유선인터넷을 위한 케이블로 공유기를 pc와 연결하는데 사용한다.
39) 땜납을 녹여 두 가지 이상의 금속을 연결하는 것을 말한다.
40) 시스템에서의 물리적 변화의 반응에 대한 원인과 결과의 지연 시간이다.

데이터로 바꿔 아두이노로 값을 받고, 구조물 구름으로 데이터를 보내어 LED 빛을 제어한다. 바텐을 이용해 무대 중앙에 구름 구조물이 공중에 떠 있게 고정하였고 바텐의 위, 아래 조종으로 공연 때맞춰 내려올 수 있게 하였다. 하모니카 연주자는 구조물 밑에 위치 했고, 오퍼레이터는 객석 뒤쪽에 위치하여 사운드와 LED를 제어하였다. [그림-20]은 공연 때 사용되는 시스템 구성이다.




[그림-20] 공연 시스템 구성

# Ⅲ. 연구기술의 작품 적용

## 1. 작품 구성

<표-8>은 작품의 전체적인 구성 표이다. 작품 <Sail>은 하모니카가 하나의 작은 배를 연상하게 하여 항해하는 느낌을 음악에 담으려 하였다. A-B-C-A'로 이루어진 구성으로 A 파트의 4/4박자에서 B 파트 rubato로 바뀌는 부분을 통해 음악의 빠른 부분으로 발전되기 위한 구성을 했고, C 파트의 리듬 부분이 빨라지는 느낌의 효과를 주도록 6/8으로 박자를 바꾸었다. A' 파트는 다시 4/4박자로 변박 되며 긴장감이 해결되는 구성으로 진행된다. 작품 전체의 스토리는 컴퓨터의 사운드 프로세싱의 소리와 하모니카의 오리지널 소리가 자연스럽게 합쳐지도록 하였다. 하모니카 연주자가 음악의 해석이 쉽고 연주에 몰입할 수 있도록 작품의 구조와 의미를 설명했다.

**<표-8> 작품의 구성**

| 구성 | A | B | C | A' |
|---|---|---|---|---|
| 시간 | 00:00~03:22 | 03:22~04:33 | 04:33~05:38 | 05:38~07:20 |
| 의미 | 배의 출항, 나아감 | 고요한 바다 | 폭풍을 만난 배 | 배의 입항 |
| 박자 | 4/4 | rubato | 6/8 | 4/4 |
| 주요 사운드 프로세싱 | reverb, granular synthesis, phase vocoder | phase vocoder | cross~+ granular synthesis | granular synthesis, flanger |

## 2. 작품에서의 사운드 및 LED 기술 적용

### 1) A 파트

A 파트는 배가 선착장에서 출항하는 것을 의미하는 부분으로, 하모니카의 음을 길게 연주하여 표현했다. reverb의 값이 서서히 많이 출력되도록 하면서 오리지널 사운드에 공간감을 부여했다. 또한, granular synthesis를 통해 하모니카의 원본 소리보다 저음의 소리가 나오도록 변형시킨 사운드를 사용하여 풍부한 음향효과를 구현했다.

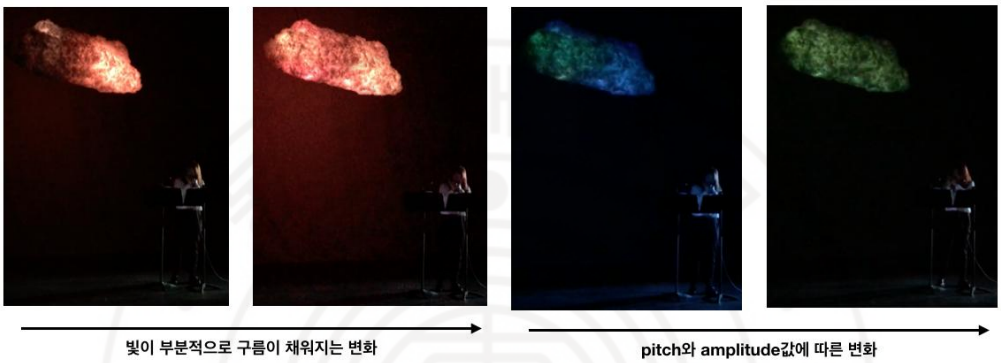
A파트 중간 부분은 음의 길이를 짧게 하고 화음을 연주하여 첫 부분과는 다른 느낌을 표현하였고, 스윙 리듬(swing rhythm)[41]으로 발전되어 박자가 조금씩 빨라지는 것처럼 의도했다. granular synthesis로 변형된 높은 음정과 낮은 음정은 화음의 느낌을 주었고, glissando 주법과 phase vocoder의 실시간 reverse를 이용해 음계가 올라갔다 내려오는 효과를 주었다

**<표-9> A 파트 구성**

| 파트 | A | |
|---|---|---|
| 시간 | 00:00~01:32 | 01:30~03:22 |
| 사운드 프로세싱 | **granular synthesis** (grain pitch = 0.25) | **granular synthesis** (grain pitch = 0.25, 1) **phase vocoder** (playback rate = -0.5) |
| LED | section 1 | section 3 |

41) 규칙적으로 반복되는 리듬감을 말하며, 당김음을 이용하여 원래의 박자에 비해 박자가 빠르거나 느린 리듬을 말한다.

LED의 부분은 section 1과 section 3을 사용했다. section 1은 음악의 시간에 맞춰 빛이 서서히 구름이 채워지는 것을 나타냈고, section 3는 연주의 짧고 리듬 적인 부분을 음정과 음량 값으로 실시간 색감을 표현하였다. [그림-21]은 A 파트의 구름 변화 사진이다.




[그림-21] A 파트 구름 변화

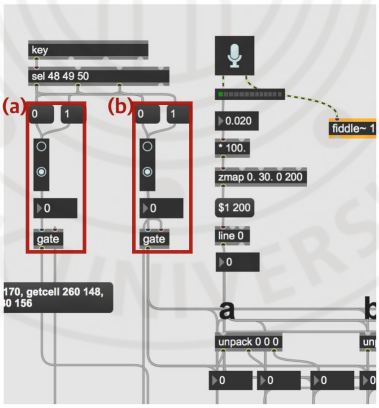
### 2) B 파트

B 파트는 rubato를 통해 음악 전체의 안정감을 주었고, 다음 파트의 리듬과 자연스러운 연결을 위해 템포(tempo)가 서서히 빨라지게끔 했다. cross~오브젝트에 1200Hz보다 낮은 주파수를 설정하여 granular synthesis의 저음 부분을 이용해 날카로운 고음 부분을 감쇄시켜 사용하였으나 낮은 영역의 주파수만 출력되면서, 사운드가 많이 변형되었다. 이를 grain separation과 variation의 값을 조절하여 부자연스러움을 해소했다. phase vocoder를 통해 1초 뒤에 사운드가 reverse되어 출력되게끔 설정하였고 이를 통해, 이중주의 느낌을 구현했다.

<표-10> B 파트 구성

| 파트 | B | |
|---|---|---|
| 시간 | 03:22~04:33 | |
| 사운드 프로세싱 | **phase vocoder** (playback rate = -0.5)<br>**cross~+granular synrthesis**<br>(grain separation = 8,<br>grain separation variation = 3,<br>grain pitch = 0.25) | |
| LED | section 2 | section 3 |

B 파트의 section 2로 전환하기 위해, 데이터 입력에 대한 설정을 키보드로 변경하였다. 다음의 [그림-22]와 같이, 다양한 데이터를 작품에 활용하기 위해 (a)의 부분은 영상의 데이터를 출력하고 (b)의 부분은 실시간으로 들어오는 사운드의 데이터를 출력하도록 했다.




[그림-22] LED 스위치의 패치

section 2의 LED는 B 파트 음악의 느려진 부분과 어울리리도록 영상데이터를 이용해 빛을 표현하였고, section 3의 LED는 C 파트의 리듬과 연결하기 위해 사운드 데이터를 이용해 빛을 표현하였다. 이때, C 파트에 긴장감과 고조된 분위기를 연출하기 위해 section 3의 LED에서 사용된 사운드 데이터(음정과 음량)의 값을 적게 주어 section 3의 빛의 변화 값을 크지 않게 하였다. 고요한 바다의 모습과 폭풍이 오기 전 몰려오는 구름을 연상하게 표현하였다.[그림-23]


**[그림-23] B 파트 구름의 변화**

### 3) C 파트

C파트로 진행될 때 4/4에서 6/8으로 변박되어, 음악이 발전된 양상으로 흘러가는 파트이다. 이 부분은 먹구름이 몰려와 바다의 움직임이 강해지는 것을 표현하였다. 하모니카의 텅잉 주법을 이용하여 작품의 극적인 분위기를 연출했고, cross~오브젝트를 사용해 각각 1200Hz, 1000Hz, 800Hz 이하로 나눠 주파수를 출력했으며 granular synthesis와 연결되어 저음, 중음, 고음을 출력했다. 각각의 주파수

영역으로 나눠 사운드 변형을 한 이유는 저음은 리듬이 길게 따라오게 표현하였고, 중음은 고음과 저음역에는 없는 중간 부분의 화음을 채워주기 위해 사용하였고, 고음은 하모니카의 음색과는 다른 소리가 연주하듯 표현하였다. 또한, 고음으로 출력되는 부분은 grain pitch의 값을 2~4로 실시간으로 제어하여 음악의 고조되는 부분을 강조했다.

**<표-11> C 파트 구성**

| 파트 | C |
|---|---|
| 시간 | 04:33~05:38 |
| 사운드 프로세싱 | **cross~+granular synrthesis**<br>(주파수 1200Hz이하, grain separation = 8,<br>grain separation variation = 3, grain pitch = 0.25)<br>**cross~+granular synrthesis**<br>(주파수 800Hz이하, grain separation = 10,<br>grain separation variation = 5, grain pitch = 0.5)<br>**cross~+granular synrthesis**<br>(주파수 1000Hz이하, grain separation = 0,<br>grain separation variation = 0,<br>grain pitch = 2~4(컨트롤러조절)) |
| LED | section 2 |

C 파트의 LED는 영상데이터를 활용하여 강한 색감과 움직임을 표현했다. 변박과 빠른 리듬을 통해 음악이 발전되면서 LED의 빛 변화 속도도 음악의 전개에 맞춰, 빠르게 움직이도록 했다. 이는 폭풍과 번개가 치듯 구름의 빛이 같은 역동적인 이미지를 표현한다.[그림-24]


[그림-24] C파트 구름의 변화

### 4) A' 파트

클라이맥스(climax)인 C 파트가 끝나고 난 뒤, A' 파트로 진행되며 6/8 리듬이 4/4로 다시 돌아간다. 이 부분은 항해하는 배가 힘든 시기를 지나 도착하는 것을 표현하였고 delay와 flanger의 효과를 이용하여 하모니카의 악기 소리가 아닌 바람 소리처럼 표현하였다. flanger의 rate와 depth의 실시간 제어로 위상을 변화시켜 바람 소리의 변화를 주었으며 granular synthesis의 grain pitch 값을 4로 주어 바람 소리와 함께 변형된 사운드가 출력되도록 했다.

<표-12> A' 파트 구성

| 파트 | A' | |
|---|---|---|
| 시간 | 04:33~07:20 | |
| 사운드 프로세싱 | **granular synthesis** (grain pitch = 0.25, 4)<br>**flanger** (Left delay = 300, Right delay = 500,<br>flanger rate = 0.~0.6, depth = 0.~0.1) | |
| LED | section 4 | section 1 |

A’ 파트의 LED는 section 4와 1로 나누어 볼 수 있다. 잔잔한 바다와 배의 평화로움을 표현하며 곡의 마무리 부분을 명확하게 전달하기 위해, section 4와 1에서는 어울리는 빛 색상의 수치를 직접 입력하여 사용하였다.[그림-25]


[그림-25] A’ 파트 구름의 변화

하모니카 연주에 Max를 통한 사운드 프로세싱의 음향효과는 악기 주법의 특색을 두드러지게 해주었고, 음악의 표현을 풍성하게 해주었다. 하모니카의 선율과 리듬을 연주할 때, 악기 한 대의 연주를 granular synthesis의 사용으로 저음과 고음을 보강하여 여러 대의 하모니카가 연주하는 사운드를 만들어 낼 수 있었다. 또한, phase vocoder를 통해 하모니카 글리산도 주법의 연주에 reverse가 된 음계가 연장되는 소리가 들릴 수 있게 하였고, delay와 flanger를 통해 하모니카 소리를 바람 소리의 음향효과로 바꾸어 새로운 사운드를 만들 수 있었다. 악기의

본래의 주법과 음색뿐만 아니라 사운드 프로세싱의 제어로 혼합된 사운드는 작품의 다양한 표현을 효과적으로 만들 수 있다.

악기와 연동된 LED 설치작품은 빛을 통해 작품의 전체적인 구성에 효과적으로 표현하였다. 실시간 사운드의 음정과 음량 값을 통해 빛의 색을 만들었고, 빛의 움직임을 영상으로 제작하여 음악 구성에 이해를 도왔다. 악기와 설치작품의 연동은 관객들이 음악과 빛이 상호작용하는 것을 느낄 수 있도록 하였고, 빛의 청각화를 구현할 수 있는 멀티미디어 음악 작품을 표현하도록 했다.

# Ⅳ. 결 론

본 연구는 하모니카 연주의 실시간 사운드 프로세싱을 통해 하나의 악기 연주만으로 만들 수 없는 소리를 구현하고 사운드의 음정과 음량 값을 통해 LED 빛과 연동되는 멀티미디어작품 제작에 관한 연구이다. 작품 <Sail>은 Max를 사용하여 실시간 사운드 프로세싱을 통해 제작되었으며 악기의 특성을 강조하기 위해 granular synthesis, phase vocoder, flanger, delay를 활용하였다. 사운드 프로세싱을 작품에 적용하면서 악기의 주법, 음역, 음색의 표현이 다양해졌고 기존의 하모니카 음악과 차별을 둔다. Max를 통해 들어오는 사운드의 데이터를 아두이노로 전송하여 LED가 부착된 구조물에 출력하였고, 실시간으로 전송되는 음정과 음량 값의 데이터는 LED의 색감을 표현한다. 이때, 구조물의 스트립을 부분적으로 나누어 부착함으로써, 빛의 빠르기를 영역별로 다르게 적용시켰다. 따라서 빛의 역동적인 표현이 가능하게 하였다. 그러나 LED 픽셀 한 개씩의 개별적인 제어가 아닌, 한 그룹의 스트립 단위를 제어하였기 때문에 더욱 세밀한 표현이 부족하였다. 또한, 프로그램 사이의 데이터 전송이 불안정한 경우 LED의 값이 변동하는 현상이 발생하여 제어에 어려움이 있었다.

이번 연구를 통해 발견된 문제점은 두 가지이다. 첫 번째, 본 공연처럼 작품의 모든 요소를 실시간 제어했을 때는 데이터 전송을 원활하게 하고 불필요한 코딩들을 수정하여 컴퓨터가 느려지는 요인들을 줄여야 한다. 연산처리가 느려지면서 실시간 제어에 위험요소가 따르기 때문이다. 더 나아가 가장 효과적인 방식은 작품의 모든 요소를 자동화하는 것이다.

사운드 프로세싱과 LED 연동의 자동화된 시스템을 기반으로 실시간 제어에 대한 부담을 줄이고 실수 및 여러 오류 들을 줄일 수 있을 것이다. 두 번째, LED 픽셀 단위의 제어와 불안정한 데이터 값이다. LED의 설치작품은 공연장에 비해 크기가 작았고, 넓은 스트립의 단위는 작품의 다양한 표현이 부족했다. 빛의 섬세하고 다양한 움직임을 위해서 각 픽셀의 제어에 관한 연구가 추가적으로 이루어져야 한다.

작품 <Sail>은 음악에 따른 빛의 움직임을 그려낸 작품으로, 시각만 활용했던 선행 연구 때보다 더 발전되었으며 작품의 완성도를 높일 수 있었다. 시·청각의 두 가지 미디어를 통해 작품을 구성하고 있기 때문에 창작자의 의도와 작품의 주제를 명확하게 표현할 수 있었으며 작품의 전달력이 향상되었다. 이를 통해, 관객이 작품을 쉽게 이해할 수 있도록 했고 향후, 앞서 언급된 두 가지 문제점들을 보완하는 연구를 통해 예술에 대한 다양한 형태를 제시하고 현대 예술에 대한 접근성을 높이고자 한다.

Keyword(검색어)
컴퓨터음악(computer music), Max, Arduino,
실시간 사운드 프로세싱(real-time sound processing),
멀티미디어음악(multimedia music), 미디어아트(media art)

E-mail : sara4489@naver.com

# 참 고 문 헌

## 1. 단행본


- 김근호, 「오디오 용어사전」, (새녘출판사, 2013)

- 김영민, 「사운드 디자인을 위한 맥스」, (Real Lies Media, 2017)

- 이석원, 「음악음향학」, (심설당, 2003)

- 최연희, 「현대의 예술과 미학」, (서울대학교출판부, 2007)

- V. J. Manzo, 「Max/MSP/Jitter for music」,
  (Oxford University, 2011)

- Curtis Roads, 「The Computer for music」,
  (MIT Press, 1996)


## 2. 참고논문


- 강현우, 「인도음악 연구를 통한 인터랙티브 멀티미디어음악 제작 연구」 (동국대학교 영상대학원 멀티미디어학과, 2017)

- 이도경, 「피아노 연주를 통한 실시간 오디오-비주얼 작품 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 멀티미디어작품 제작 연구」 (동국대학교 영상대학원 멀티미디어학과, 2019)

- 최아영, 「피아노 연주의 실시간 사운드 프로세싱을 이용한 멀티미디어작품 제작 연구」 (동국대학교 영상대학원 멀티미디어학과, 2019)

### 3. 웹사이트

- CNMAT: external Max object, OSC
http://cnmat.berkeley.edu/

- Arduino
https://www.arduino.cc/

- Max
https://cycling74.com/

# ABSTRACT

## A Study on the Production of Multimedia Music using LED and Real-time Sound Processing for Harmonica

(focus on Multimedia Music <Sail>)

Oh, A Young

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

<Sail> is a study on the production of multimedia using real-time sound processing and LED of harmonica performance. The interaction of the sound in real time and the response of the LED installations made the auditory of the light a multimedia work.

The Max program produced sound real-time processing. Harmonica's real-time sound processing was applied to music by creating sound effects through granular synthesis, phase vocoder, flanger, and delay. With the renewal of the instrument's rhythm, range, and timbre, it

differentiates it from traditional harmonica music. The sound data coming through Max was sent to the Arduino and output to the LED installation. The pitch and volume of the sound are used to express the color of the LED light in real time, and the movement is expressed by Jitter's image control. The LEDs in the installation show the dynamic movement of the strips.

In addition, the control of the pixels of the LED and the study of unstable values are required.

As a result, the audience could feel that the music and the LED installations interacted. The integration of music and installations is controlled through a computer system and works reliably without data delays or errors.

# 부록 : 첨부 DVD 설명

1. <Sail> 공연 영상 : 2019년 11월 16일 이해랑 예술 극장 공연

2. <Sail> patch : 작품에 사용된 Max 패치 폴더

3. <Sail> Arduino : 작품에 사용된 아두이노 코딩