|  |
| --- |
| **붙임 3** |



2018 학년도 제 2 학기

제목 : 머신러닝을 통한 드럼악보 추출

홍주경(2012311332), 박동준(2013311872)

2018 년 11 월 07 일

지도교수: 윤희용 교수님 ( 서명 )

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 계획(10) | 주제(20) | 개념(20) | 상세(30) | 보고서(20) | 총점(100) |
|  |  |  |  |  |  |

|  |
| --- |
| \* 지도교수가 평가결과 기재  ○ 보고서 작성시 본 안내 텍스트상자 삭제 바랍니다.  ○ 평가표는 수정하지 않습니다.  ○ 본문 쪽번호는 필수입니다.  ○ 남의 그림을 복사하여 사용하지 않습니다. 본인이 직접그림을 그립니다.  ○ 모니터화면 갈무리는 가능한 사용 자제합니다.  ○ 그림에 사용된 폰트는 본문 폰트와 일치해야 합니다.  ○ 그림의 최대크기는 본문 쪽의 반으로 제한합니다.  ○ 그림 및 표의 설명(캡션)은 필수입니다.  단, 표는 위에 그림은 아래에 번호와 함께 설명을 답니다.  ○ 원천코드는 부록에 포함시킵니다.  ○ 최종보고서 분량은 총 30페이지 내외로 작성합니다.    평가 배점  계획 (10)  무리 없는 진행.  주제 (20)  신규성, 진보성, 창의성, 현실성  개념설계 (20)  주제에 명시된 바대로 필요한 기능이 모두 포함되었는가?  상세설계 (30)  계획에 명시된 바대로 결과물과 일치하는가?  각각의 practical constraints에 기술된 사항을 만족하는가?  보고서 (20)  짜임새 있는 정리 및 표현 |

**Contents**

0. 요약 …………………………………………………………………………………………………………………… 5

1. 서론…………………………………………………………………………………..………………………………… 6

1.1. 제안배경 및 필요성 …………………………………………………………………………………………………….... 6

1.2. 연구/논문 작품의 목표 ………………………………………………………………………………………………… 6

1.3. 연구/논문 작품의 전체 Overview ………………………………………………………………………………….. 6

1.3.1. Sound Signal Extraction and Drum Signal Extraction …………………………………………… 6

1.3.2. Machine Learning …………………………………………………………………………………………….. 6

1.3.3. Visualize with Musical Score ……………………………………………………………………………… 6

1.4. 선행연구 및 기술현황 ……………………………………………………………………………………………………. 6

1.4.1. Hum On ………………………………………………………………………………………………..……………………... 7

1.4.2. Machine Learning …………………………………………………………………………..……………………………… 8

1.4.3. Artificial Neural Network …………………………………...…………………………………………………………… 8

1.4.4. Decision Forest ……………………………………………………………………………………………………………... 8

1.4.5. MIR(Music Information retrieval) …………………………………………………………………………………….. 9

1.4.6. Visualization …………………………………………………………………………………………………………………. 7

2. 관련연구 ……………………………………………………………………………………………………………. 8

3. 제안 작품 소개 ………………………………………………………………………………………………… 10

3.1. 시스템 구성 ………………………………………………………………………………………………………………… 10

3.1.1 Ubuntu 16.04LTS ……………………………………………………………………………………………. 10

3.1.2 Python v3 ………………………………………………………………………………………………………. 11

3.1.3 TensorFlow ……………………………………………………………………………………………………... 11

3.2. 이론적 배경 ……………………………………………………………………………..………………………………….. 12

3.2.1 Datasets for Drum Sounds ……………………………………………………………………………….. 12

3.2.2 LibROSA …………………………………………………………………………………………………………. 12

3.2.3 Scikit-Learn ………………………………………..…………………………………………………………… 13

4. 구현 및 결과분석 …………………………………………………………………………………………..… 14

5. 결론 및 소감 ………………………………………………………………………………………………….… 16

6. 참고문헌 ………………………………………………………………………………………………………….. 18

■ 요약 (추가할내용 없음)

Recently, drum, the most popular and important percussion instrument in modern music, is an instrument that many people want to learn. However, many people are suffering from difficulties of making drum scores. Moreover many people feel that there are not enough materials about drum score, and also people found that making drum scores are labor-intensive work. a ppeal for lack of materials to find or make music.

For those who are experiencing this difficulty, we want to create a product that extracts drum scores from voice files. The working principle of the work is as follows. First, from the voice file with drum sound, we classified each musical instruments’ sound signal and extracted the data set of drum sound by using MIR(Music Information Retrieval) Technology After that, by applying one of machine learning algorithm and multiclass classification method, categorize classified drum sound by 7 kinds. Finally visualize the categorized data with the drum score.

■ 서론 (6페이지 내외)

연구논문/작품 전체 overview

* 여기에 결과에대한 내용만 추가하면 됨

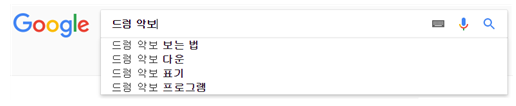
**1.1. 제안배경 및 필요성**



**<Figure 1>** Infographic of Survey Reportfrom JakPat:

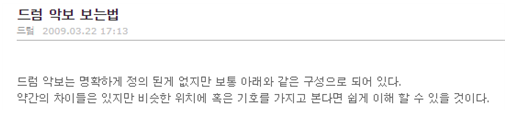
People’s Interest in Playing a Musical Instrument

Drums are the most popular and important percussion instrument in contemporary music. According to the <Figure 1>, a random survey of internet site JakPat, 49.18% of the survey participants replied that they have interest in percussion, including drums. As such, drums are not the only thing for specialized people, and many people are learning drums as a hobby.



**<Figure 2>** Related keyword of “drum score” search in Google Korea

Despite of these popularization of drums, however, drummers including hobbyists faced large obstacles. According to <Figure 2>, when we search drum score in Google Korea, the first thing in the auto-complete related query is "How to view drum score". Even in the lecture about “how to view drum score”, as in <Figure 3>, says "There are no clearly defined method to make drum score.". As such, reading and making drum score is a quite cumbersome work, and so if you are not a major in drums, writing your own drum score is a very difficult and demanding work. So when people are looking for a piece of drum score that they want, or when they want to write a new piece of drum score, they suffer from the lack of materials.



**<Figure 3>** Lecture about how to view drum score

**1.2. 연구논문/작품의 목표**

In order to make it easier for many people who are experiencing this difficulty, we want to create a program that can extract the drum music score by inputting the music file. And, with this work, we hope that many drummers will be able to find and make drum score easier, and hope to be able to lead a more convenient cultural life.

**1.3 연구 논문/작품 전체 Overview (+ 작품결과에 대한 내용 - 2page+@)**

**1.3.1. Sound Signal Extraction and Drum Signal Extraction**

It is the process of getting the data sets of the voice file or the sound source prior to the machine learning. Refer to existing MIR-based technologies and understand how to extract sound signals from audio sources by using matlab program through internet research. In addition, the extracted signal undergoes secondary classification with 7 kinds for example, cymbals, snare drum, hi-hat, etc.. according to the drum sound. Classification will depend on the frequency of the drum part.

**1.3.2. Machine Learning**

Before letting the computer to learn the actual dataset, understand the various machine learning modeling techniques first and find the appropriate technique to use. Moreover by using Python and Matlab, we will study how to apply machine learning technique to the computer. In addition, when we input the real data set to let computer to learn, we will use an appropriate method and programming language. Finally, calculating the accuracy by applying various machine learning methods.

**1.3.3. Visualize with Musical Score**

According to the result from 3.3, visualize the data from result with open source software which help user to draw musical score. Prior to use that , we need to first study and understand the instructions of that software. Then we can finally draw a musical score by inputting data set with coding.

**1.4 선행연구 및 기술현황**

**1.4.1. Hum On**

There is an application called 'Hum On'. It is a simple concepted application that turns humming, or hum, into a musical note. However this simple concept contains really complex technique. The sound of humming is different for person by person. In addition, even the same sound of the same person is different from the sound of other situations. The 'Hum On' is the application that transfers this to the musical score.

Machine running is the basis for making this difficult technology available. 'Hum On' analyzed humming 'Big Data' by machine learning technology and applied the technique of reading user intention. With this technology, you can turn humming into musical score as well as other sounds.

**1.4.2. Machine Learning**

Machine learning means let machine to learn something. In other words, even if a person does not explicitly direct logic to a computer, it means that the computer can 'learn' through a large number of data sets and automatically solve the problem through the learned 'experience'. Like that, machine learning is a field of research that develops algorithms that allow machines to "learn" from data and to perform actions that are not explicitly specified in code.

**1.4.3. Artificial neural network**

Artificial neural network, which is a statistical learning algorithm of machine learning inspired by neural networks of biology. An artificial neural network refers to the entire model that has artificial neurons (nodes) that form a network due to defects of the synapses, which change the binding strength of the synapses through learning and have problem solving ability.

Artificial neural networks include teacher learning that is optimized for problems by inputting teacher signals (answers), and companion learning that does not require teacher signals. In the case of this work, the actual music and its drum score can be used, so teacher learning will be used.

**1.4.4. Decision Forest**

Decision Forest is a forecasting model which connects observed value with desired value. It is a kind of forecasting modeling method using decision tree in machine learning. Classification tree is a tree model with finite target variable. In this tree structure, a leaf node is a class label and branch is a logical product of characteristics of class label.

Decision tree is visual and explicit. It is used to express decision making process and decision made or data itself. In this project, artificial neural network will be replaced if decision forest is more precise than artificial neural network.

**1.4.5. MIR(Music Information Retrieval)**

MIR is a technology which analyzes and extracts rhythmic information of audio contents automatically and visualize it. MIR is a kind of Digital signal process(DSP), and it is studied more these days. MIR analyzes specific vectors made by parameterizing audio signal, with techniques such as pattern matching. In plain language, MIR changes music sound to 'sound signal'. Music is just a gathering of some sounds. However, if the music is changed to electric signal, it becomes information. It can be sole signal or splited signal, to be sent to someone. In now, especially in web or mobile, MIR is applied to common applicaion.

**1.4.6. Visualization**

There are not many programs which can draw music score with computer. Moreover, few programs provide circumstance where we can use data from machine learning to draw music score.

In this project, we use a software named 'Lilypond'. Lilypond is complete open source software. It has all functions required to typing music score. Also, many users are improving it animately and can use it for free, because it is open source software.

■ 관련연구 (7페이지 내외)

연구논문/작품과 관련된 related work를 작성합니다. 국문논문 영문논문 무방합니다만, 적어도 국문 3개 이상 영문 3개 이상을 포함합니다. 각 관련된 문장에 관련된 연구 참조문헌 번호 기록바랍니다. [2]

* 지난번에 했던 것들 2페이지 ;;; ⇒ 7페이지로 늘려야함

■ 제안 작품 소개 (7페이지 내외)

실제로 구현에 대해 소개합니다. 자세하면 자세할수록 좋습니다. 이론적 배경과 시스템구성은 필수입니다.

* Flow chart, visualization 때 사용할 lilypond라는 악보그리는 프로그램 소개 (2페이지 추가 요망)

**3.1 시스템 구성**

In this part we will going to introduce our development environment. What OS, Program for coding and deep learning method did we chose.

**3.1.1 Ubuntu 16.04 LTS**



**<Figure 4>** Simple Screenshot for Ubuntu 16.04 LTS

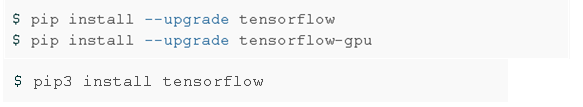
For OS, we selected Linux Ubuntu version 16.04 LTS. Figure 4 shows the simple screenshot of Ubuntu. To briefly introduce this environment, it’s code name is Xenial Xerus and is Debian GNU/Linux based. Ubuntu has several important characteristics which are first, it is optimized for personal desktop environment. It provides comfortable user interface. Second, it is free software based. So we can use this OS for free. Alse, there are several improvements from previous version which was Ubuntu 14.04 LTS. First, basic setting is now python v3. In previous version, python v2 was used. We will explain about python 3 at next stage. Furthermore, lots of applications are improved for user convenience.

**3.1.2 Python v3**

For programming language, we used Python 3 which has been growing in popularity over the last few years. This is an interpreted language so is passed straight to an interpreter that runs the code directly. This makes for a quicker development cycle because we just type in our code and run it, without the intermediate compilation step.

Recently Python is taking center stage as a machine learning language. This is because of its simplicity, it has machine learning library and it manages memory for user.

**3.1.3 TensorFlow**



**<Figure 5>** Installing Tensorflow

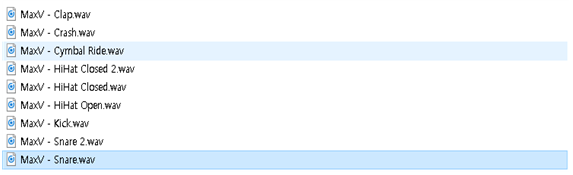
TensorFlow is an open source software library for numerical computation using data-flow graphs. It was originally developed by the Google Brain Team within Google's Machine Intelligence research organization for machine learning and deep neural networks research, but the system is general enough to be applicable in a wide variety of other domains as well.

TensorFlow is cross-platform. It runs on nearly everything: GPUs and CPUs—including mobile and embedded platforms etc.

We installed tensorflow to our desktop by following commands.

**3.2 이론적 배경**

**3.2.1 DataSet for Drum sounds**

****

**<Figure 6>** How the data are arranged

For the first thing we do for this project, is to find various Drum audio datasets for deep learning method. The more audio source file we have, the more accurate result will be given.

However it was impossible for us to get tons of drum audio files by self recording. So we found a opensource datasets in google. This dataset for drum machine contains 200 type of drum sounds which are from various drum producing company. It contains electrical, classic and vintage etc. Those drum sound data are agains divided in to numerous types which are snare, clap, cymbals, hihat and toms. Figure 5 shows how the datasets are divided and arranged. Since there were some unneeded sounds and data are not arranged, we unified the file names and left only files we needed. By those various kinds of drum sounds we can cover various type of audios.

**3.2.2 LibROSA**

LibROSA is a python package for music and audio analysis. It provides the building blocks necessary to create music information retrieval systems.

**3.2.3 Scikit-learn**

Scikit-learn is an open source tool that can be used to data mining and data analysis, built on NumPy, SciPy and Matplotlib.

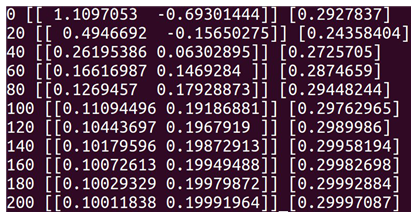
■ 구현 및 결과분석 (7페이지 내외)

소정의 결과를 소개하고 결과에 대한 분석을 적습니다.

* 지난번에 했던건 그저 데이터 분석 여기다가 데이터 분류 + 머신러닝 + visualization 쓰기 (+ 4페이지 요망)

We used Tensorflow in machine-learning. The program learned a learning dataset and used testing dataset. Datasets have datas with 20 dimension vectors. In result, the program assumed the answer with very low error rate. For example, in <figure 6>, Its error rate was below 0.001% in 200 times of machine learning with simple 2 dimension vector.

Then, we searched datasets to use in machine learning. We found 7,476 sounds by 200 drums. We used LibROSA and Scikit-learn to analysis these datasets.



**<Figure 7>** A simple Tensorflow example

First, LibROSA is a python package for music and audio analysis. It provides the building blocks necessary to create music information retrieval systems. So we used LibROSA to load drum sound datasets to the program and change them MFCC. MFCC is the most popular feature value and can be gained by MFC. The mel-frequency cepstrum (MFC) is a representation of the short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a nonlinear mel scale of frequency. MFC can be splited to this six steps:

* Cut audio signal of input time domain down to small frames.
* Compute Periodogram estimate(Periodogram Spectral Estimate) of Power Spectrum for each frames.
* Apply Mel Filter Bank to Power Spectrum of 2, then plus energy to each filter.
* Take log to every filter bank energy of 3.
* Take DCT to value of 4.
* Get Coefficients 2~13 of result of DCT.

After this 6 steps, the Coefficients is called MFCC. We assume 20 MFCCs, and if it lacks, we will increase that up to 50.

Scikit-learn is an open source tool that can be used to data mining and data analysis, built on NumPy, SciPy and Matplotlib. We will use this classifier using MFCC vectors as input by LibROSA and 4 drum sound classes as output.

In this step, we must prevent Overfitting. We don’t have to use all of 7,476, but have to use 4,476 to learn and 3,000 to test.

We will analysis, use to learn, test and use based on actual data. However, other loud sounds which is similar to drum sounds can be recognized as drum sounds. To prevent this, we are considering adding the dMFCC and ddMFCC to our learning data or make to learn the sounds of other instruments as well.

■ 결론 및 소감 (2페이지 내외)

결론을 맺습니다.

연구논문/작품을 하면서 겪었던 소중한 경험들에 대해 적습니다. 가장 중요한 부분입니다.

* 지난번에 했던거에 양을 조금더 늘리면 될 듯 함

**5.1. Works for future**

For the future we will use TensorFlow to let computer learn sampled data

(the drum signals) from audio file. Then we will make a code to classify 4

different drum sounds. For this work, we might increase the amount of drum

sound types. At the initial perspective, we decided to divide drum sounds for 4

different types cymbals, snare, toms, and kicks. However, now we realized that

there are more sounds that we need to focus. At last we will make a code to

match drum score and specific drum signal. this is also called visualization step.

The major thing in this step is to show the specific drum signal to the musical

score.

**5.2. Conclusion and Perspective**

By advancing this graduation work, we have experienced many things that we cannot experience in general undergraduate courses.

First of all, with my team member, we got together in a place and made a lot of technical discussion. This made us to increase major knowledge and learn how to solve various problems.

Moreover, since this was the first time for us to do this big project, we systematically divided the project into parts and solved them step by step. This made us to solve problem fluently.

Finally, we used pair programming method to code our project. By using this method, we shared our major knowledge and anguished specific problem together. This made us to more concentrate on our work. Plus, we can also immediately proceed the code review and help to find each other’s errors.

■ 참고문헌 (1페이지 내외)

인용한 참고문헌입니다. 참고문헌 인용은 다음과 같이 합니다 [1].