ALTERNATE TUNING CHORD FINDER

Apoorv Gupta Mihir Lovekar

Computer Science and Engineering Computer Science and Engineering

SRM Institute of Science and Technology SRM Institute of Science and Technology

Kattankulathur Kattankulathur

ap6107@srmist.edu.in ms5377@srmist.edu.in

ABSTRACT

Most beginners stick to the classic standard tuning of the guitar. Chords are essentially multiple notes played together. Chords, to be played, need the player to fret certain strings or all strings and strum. The places where the string needs to be fretted is determined by the tuning of the guitar. Essentially changing the tuning of the guitar shifts the notes left or right depending on whether the string is tuned up or down. This project aims to find chords on alternate tuning. That can be computed by taking the tuning and the chord as the input and finding all notes of the chord, eliminating the isolated notes 3 and only taking a margin of frets where all notes occur (for the human hand to play). This is achieved by combining various modules and algorithmic tasks to suit the best optimized solution. This freedom enables the player to enhance their playing by finding different chord voicings on foreign tunings. These alternate tunings are used by a lot of intermediate players since it allows them to sound unique and implement voicings that give their music an edge over the traditional chord voicings.

1. INTRODUCTION

The aim of this project is to build an application which computes the playable chord shapes on the guitar set to any alternate tuning. The user will input the desired tuning on which he/she wants to play, they will input the chord they wish to know how to play on said alternate tuning. The result will be all the playable positions of that chord on the guitar fretboard in that tuning. This is achieved by writing an intricate algorithm which computes this output in multiple steps. This project has a high algorithmic approach and the key highlight of it is the optimization of the algorithm used. The ease of which the user can find and access something which he/she would have otherwise have to find manually.

1. LITERATURE SURVEY

Regarding this a paper published by Filip Korzeniowski and Gerhard Widmer titled A FULLY CONVOLUTIONAL DEEP AUDITORY MODEL FOR MUSICAL CHORD RECOGNITION in 2016 states that in general typical harmony acknowledgment pipelines include three stages: feature extraction, design coordinating, and harmony arrangement de-coding. In the paper Ultimate Guide to Alternate Tunings by Aaron Mathies dated August 2019, he suggests the benefits of alternate tunings as - 1. Gives you admittance to new notes and sounds. 14 2. Makes new harmony shapes and finger positions. 3. Opens up your brain to better approaches for deduction and motivation. Furthermore, each tuning resembles playing an alternate instrument. Some of them will feel recognizable, while others will feel so unfamiliar that it's practically like beginning without any preparation.

In addition to finding the chords, the next step would be to recognize the chord when it is in continuation of being played. Regarding this a paper published by Filip Korzeniowski and Gerhard Widmer titled A FULLY CONVOLUTIONAL DEEP AUDITORY MODEL FOR MUSICAL CHORD RECOGNITION in 2016 states that in general typical harmony acknowledgment pipelines include three stages: feature extraction, design coordinating, and harmony arrangement de-coding. Feature extraction changes sound signs into representations which accentuate content 15 identified with harmony .

1. PROPOSED WORK

The system which we propose consists of a fully operational Application which will Compute chord shapes and positions on the guitar fretboard in any tuning taking the tuning and chord as an input.

● All the shapes formed that are not physically possible to play will be filtered out.

● User will input the chord whose position is to be searched.

● The chord will be searched in the dataset.

● Then the subsequent notes that build the chord will be retrieved.

● All of this methodology will be wrapped around in an application. The app would be designed in Adobe XD.

● The code for the application will be written in Dart (flutter) which is a powerful language designed by google, specially to make apps for both Android as well as iOS devices.

● The application is set to follow material design so it would be easy for users to interact with the application.

1. IMPLEMENTATION

1) Tuning: Guitar tunings dole out pitches to the open series of guitars. Tunings are portrayed by the specific contributes signified by notes Western music. Standard tuning characterizes the string pitches as E, A, D, G, B, and E, from most minimal (low E2) to most noteworthy (high E4). Standard tuning is utilized by most guitarists, and regularly utilized tunings can be perceived as minor departure from standard tuning. In our application we can input any alternate tuning.

2) Chord: Chords are a specific set of notes played together. They follow the naming nomenclature of . This will be taken as an input from the user so that the specific notes of the chord can be searched on a double dimensional array representing the guitar fretboard.

3) Interface: User Interface (UI) is the portion of the application where the user can interact with the application. Typically, a user must enter a chord and a tuning and, in the backend, the Alternate tuning will be processed, and the output will be displayed on the interface.

4) Note-Mapping: The double dimensional array represents the guitar fretboard. It is a slight, long segment of material, normally wood, that is overlaid to the front of the neck of an instrument. The strings run over the fingerboard, between the nut and extension. Each cell in the double dimensional array contains a note. The algorithm will map a corresponding note to each cell of the array.

5) Dataset: Dataset contains specific notes pertaining to each chord. The dataset consists of all the common chords and the notes that build those chords. Dataset will contain common chords of all 12 notes.

6) Matrix: It is a double dimensional array representing the guitar fretboard. The size of the array is (6x23).

7) Filtering: Filtering is the process in which we use the concept of sliding window where a window of (4x23) runs over the entire double dimensional array filtering out the chords which do not lie in this window. After filtering the notes that are produced are the ones which the interface will display. That is the output.

1. RESULTS

The aftereffect of project is acquiring all possible positions to play the desired chords on the guitar fretboard in any alternate tuning possible on the guitar. It will be to create a new system for the majority guitar players , enhancing there playing conditions and improving there skills.

1. CONCLUSION

Unlocking an easier approach to incorporate alternate tunings on the guitar in one’s playing. Enabling them to map out specific chords on the entirety of the fretboard to harness different voicings and fluidity.

REFERENCES

<https://guitargearfinder.com/guides/alternate-tunings/>

https://www.guitartricks.com/blog/an-introduction-to-alternate-tunings

https://www.researchgate.net/publication/252067543\_Neural\_networks\_for\_ musical\_chords\_recognition/link/541791f40cf203f155ad5b2a/download

https://dart.dev/mobile https://blog.landr.com/alternate-guitar-tunings/ https://developer.android.com/studio/intro https://www.researchgate.net/publication/252067543\_Neural\_networks\_for\_musical\_chords\_recognition

https://sci-hub.se/10.1109/mlsp.2016.7738895

https://en.wikipedia.org/wiki/Metronome Machine Learning (cs.LG); Sound (cs.SD); Audio and Speech Processing (eess.AS); Machine Learning (stat.ML) https://arxiv.org/abs/1804.01849(Filip Korzeniowski, David R. W. Sears, Gerhard Widmer)

https://www.researchgate.net/publication/283070302\_EVALUATING\_THE \_GENERAL\_CHORD\_TYPE\_REPRESENTATION\_IN\_TONAL\_MUSIC \_AND\_ORGANISING\_GCT\_CHORD\_LABELS\_IN\_FUNCTIONAL\_CHORD\_CATEGORIES

https://dart.dev/tools/sdk

https://github.com/flutter/flutter/issue