An Innovative Musical Instrument Prototype Based on Rubik's Cube: Musical Cube

Jiaying Li
Music Technology
Georgia Institute of Technology
GA 30318 USA
jli3269@gatech.edu

ABSTRACT

This paper presents an innovative musical instrument prototype, the Musical Cube, which combines a traditional third-order Rubik's cube with music-sounding software. The Musical Cube generates chords and melody according to preset rules based on people rotating Rubik's cube. The music generated by the cube is related to the scrambled state of the cube, and different scrambled states correspond to different musical rhythms and tracks. For Rubik's Cube beginners, it is a powerful auxiliary tool that can quickly help people get familiar with the solution algorithms of Rubik's Cube. For musicians, it is a very helpful improvisation tool, and the millions of states of Rubik's Cubes provide more possibilities and fun for improvisation. The function of the Musical Cube is evaluated by its performer and audience, including the perception and usability. The evaluation result is very encouraging, and can fully reveal its future prospects and potential.

Keywords

Computer music, digital musical instruments, Rubik's Cube

1. INTRODUCTION

Invented in the late 1970s by Erno Rubik from Hungary, Rubik's Cube is one of the most famous 3D puzzles in the world. The most common version is a three-order Rubik's Cube, which consists of a 3x3x3 cube, with different colors on the surfaces of sub-cubes, or cubies. Any 3x3x1 plane of the cube can be rotated or twisted 90, 180, or 270 degrees relative to the rest of the cube. In the initial state or the solved state, cubies of each side kept the same color. A Rubik's Cube can be scrambled by making a number of random rotations, and there are more than 4×10^{19} different states that can be reached (Korf, 1997).

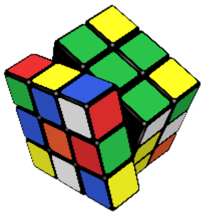

Fig 1: A standard 3x3x3 Rubik's Cube

The Musical Cube presented in this paper is designed based on a standard 3x3x3 Rubik's Cube. A GUI (graphical user interface) is designed to help interaction.

Section 2 (Design and Implementation) introduces the detailed design, implementation, and evaluation of the Musical Cube software. Section 3 (Performance Evaluation) describes the methodology of measuring the performance of the Musical Cube software in detail and the improvement suggestions for the software purposed by performers and the audience.

2. DESIGN AND IMPLEMENTATION

In this part, the design, implementation, and algorithm evaluation of the Musical Cube will be introduced.

2.1 Implementation

The implementation of the Musical Cube can be separated into three parts: Rubik's Cube implementation, sounding algorithm, and GUI design. The software is written in Python language.

2.1.1 Rubik's Cube Implementation

A standard 3x3x3 Rubik's Cube has six sides, and each side consists of nine cubies of the same color. The rotation of the Rubik's cube causes a change in the position of the small pieces, which is essentially an alternation of the positions of the permutation groups.

Using this principle, we can complete the program design of the Rubik’s Cube. It is worth noting that in the expression of the Rubik’s Cube formula, each different rotation has a fixed letter representation, as shown in Figure 2.

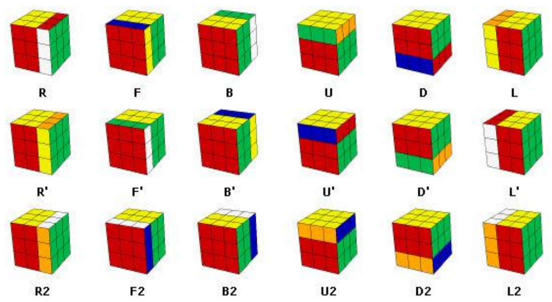

Fig 2: Rubik’s Cube notation

A standard 3x3x3 Rubik’s Cube has fixed colors on each side. We use an array function to define a Rubik’s Cube in programming.

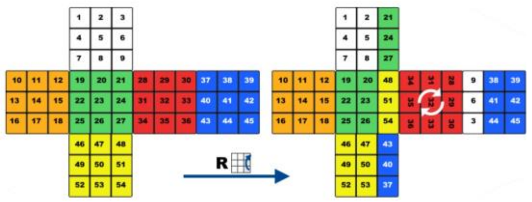

Fig 3: Basic operation (permutation) of R

2.1.2 Sounding Algorithm

The preset vocal algorithm is divided into two different tracks: the melody track and the chord track.

After scrambling the Rubik’s Cube, each side of the Rubik's Cube will be chaotic to varying degrees. Note the location of every cubie on each side as $C(i,j,k)$, where i represents the color, j represents the row, and k represents the column. If the color of $C(i,j,k)$ doesn’t satisfy the initial state, then the difference $D(i,j,k) = 1$. Otherwise, $D(i,j,k) = 0$. The chaotic index K can be written as:

$$K(i) = \sum_{j,k} D(i,j,k)$$

It’s easy to prove that $K(i) \in \{0,1,2,\dots,8,9\}$.

Table 1 shows a mapping relationship between the chaotic index K and pitches. To distinguish between melody and chord in practical application, the notes of the melody are in four-line octave (for example, C_4) and the notes of chords are in three-line octave (for example, C_3).

The duration for each note is calculated by the summation of the chaotic index of three random colors in a scrambled Rubik’s Cube.

Chaotic Index	Pitch
0	C
1	D
2	E
3	G
4	A
5	F
6	B
7	C# or D#
8	F# or G#
9	A#

Table 1: Mapping relationship between the chaotic index and pitches

The music tracks are generated by the music21 toolkit (Cuthbert & Ariza, 2010). The melody track uses electronic guitar timbre, and the chord track uses piano timbre.

There are two modes in the Musical Cube:

- A real-time player;
- A music downloader.

The real-time functions in the music21 toolkit are used to implement the real-time player, and the music stream is recorded and combined together to implement the downloader.

2.1.3 GUI design

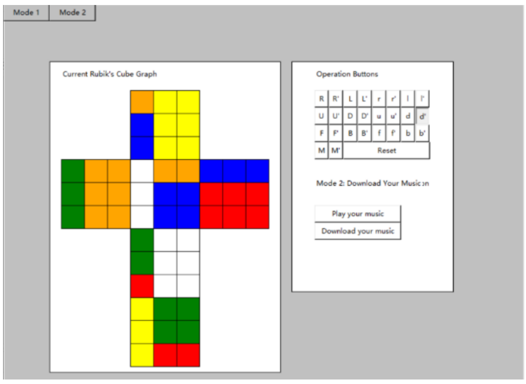

Fig 4: Musical Cube

For the convenience of users, a visual interface is essential. The interface mainly consists of three parts: control buttons, Rubik’s Cube display box, and an information text window. The GUI is designed using the Tkinter toolkit.

2.2 Algorithm Evaluation

As an improvisational instrument prototype with real-time performance capabilities, immediacy and speed of response are the most important metrics.

During the testing process, we found that if the instant music playback is not added, the rotation response of the Rubik's Cube is very fast, and the time complexity of the algorithm is $O(1)$, which shows quite high superiority. However, after adding the use and instant play function of the music21 toolkit, when encountering some complex functional relationships, there will be a certain degree of lag. Due to the use of the external toolkit, we can temporarily determine that the reason for the lag of the algorithm is caused by the slow response speed of the music21 toolkit.

To test this conjecture, we replaced the soundtrack with the simplest fixed-frequency trigonometric sound wave and got a fast-response software. But unfortunately, the single timbre of the trigonometric function cannot bring a good creative experience, which is a problem that we need to weigh and consider in future improvements.

3. PERFORMANCE EVALUATION

In this session, there are three aspects of performance that need to be considered:

- Musical performance;
- Accessibility and usability;
- Interestingness.

3.1 Experiment Design

3.1.1 Musical Performance

Play improvisations created by Musical Cube to the audience, and record their grades on a scale, which ranges from 1 star to 5 stars. Also, play improvisations created by contact microphones, and record their gradings. If the grades for Musical Cube are better than the Contact Mic, then the Musical Cube is a successful improvisation instrument.

3.1.2 Accessibility and Usability

Teach the participants basic Rubik's Cube terminology and ask them to use the Musical Cube. Record the time it takes for them to become proficient.

3.1.3 Interestingness

Ask participants in the experiment to rate the interestingness of the Musical Cube on a scale, which ranges from 1 star to 5 stars.

3.2 Experiment Procedures

A total of 8 participants of different ages were invited to join in this experiment. Before the experiment began, two 30-second audio clips were generated using the Musical Cube and Contact Mic, respectively.

Step 1: Play two audio clips to the participants and ask them to rate them using a scale separately. Very ugly corresponds to 1 star, very pleasant corresponds to 5 stars.

Step 2: Show the participants the Rubik's Cube terminology shown in Figure 3, and ask them to freely explore the use of the Musical Cube software. Record the time it takes them to become proficient in all the corresponding functions.

Step 3: Ask participants to rate the interestingness of the instrument. Very boring corresponds to 1 star, very interesting corresponds to 5 stars.

3.3 Results

From the final results, we can clearly see that for most people, Musical improvisations sound better than Contact Mic improvisations.

The average time people need to learn to play the Musical Cube is within 10 minutes, which shows that it has very high accessibility and usability.

Most people think the Musical Cube is "interesting", therefore, we can consider that the Musical Cube is a successful instrument prototype.

Age	Cube/Mic	Time	Interestingness
12	5/2	5 min	2
73	3/1	20 min	3
41	2/2	3 min	5
39	4/2	5 min	5
50	4/3	17 min	4
22	5/3	3 min	5
23	5/1	2 min	4
23	2/3	5 min	5

Table 2: Results of the performance evaluation

4. CONCLUSION

We present here the Music Cube - a new innovative computer instrument that allows users to play by rotating the Rubik's Cube to correlate the state of the Rubik's Cube with real-time playback. The algorithm of real-time playback is operated by the preset program of the Rubik's Cube Chaotic Index, mainly using music21 to add timbres and tracks.

The music cube has a graphical user interface. Users can observe the current state of the cube on the displayed images of Rubik's Cube on the GUI and listen to the improvisation created by the real-time model, or download the music clips that have been created. The Musical Cube has two different modes, and during the switching of each mode, there is text prompting to help users interact.

Finally, we also evaluate the Music Cube taking into account the performer's perspective and the audience's perspective. These evaluations show that the Music Cube has very high accessibility and interestingness as an improvisational instrument, and these evaluations also provide useful information for future development. We hope that in the future work, the music cube can achieve faster response, more options for timbres, and optimization of algorithms.

5. ACKNOWLEDGMENTS

Special thanks to all participants who kindly accepted to take part in the experiments.

6. REFERENCES

Korf, R. E. (1997, July). Finding optimal solutions to Rubik's Cube using pattern databases. In AAAI/IAAI (pp. 700-705).

Simon, H. B. (2015). Music as medicine. The American journal of medicine, 128(2), 208-210.

Rokicki, T., Kociemba, H., Davidson, M., & Dethridge, J. (2014). The diameter of the Rubik's cube group is twenty. siam REVIEW, 56(4), 645-670.

Polfreman, R., & Oliver, B. (2017). Rubik's Cube: Music's Cube.

Mannone, M., Kitamura, E., Huang, J., Sugawara, R., Chiu, P., & Kitamura, Y. (2019). CubeHarmonic: a new musical instrument based on Rubik's cube with embedded motion sensor. In ACM SIGGRAPH 2019 Posters (pp. 1-2).

Roads, C. The Computer Music Tutorial. MIT Press, 1996.

Burtner, M. (2002). The metasaxophone: concept, implementation, and mapping strategies for a new computer music instrument. Organised Sound, 7(2), 201-213.

Lee, S. W., & Freeman, J. (2013). echobo: A mobile music instrument designed for audience to play. Ann Arbor, 1001(48109-2121), 10-44.

Cuthbert, M. S., & Ariza, C. (2010). music21: A toolkit for computer-aided musicology and symbolic music data.