An Inexpensive Real-time Music Beat Tracking System Using Open Source Hardware

He Hu

ABSTRACT

In this paper, we present a low-cost (less than $5) real-time music beat tracking system using Arduino open source hardware. The beat tracking system uses KY-037 sound sensor to read sound signals from Arduino’s #1 analog input pin, and control the flag LED and beat LED through Arduino’s #7 and #8 digital output pin. With an adjustable threshold parameter, the music beat tracking system will constantly try to recognize the music beat interval when playing music. A red LED (beat LED) will be blinking according to the beat when the music beat is recognized successfully. The music beat tracking system is open sourced and the hardware design and software startup code for the music beat tracking system is available for public access through the free online service.

INTRODUCTION

Study suggests that for students, attaining knowledge in Science, Technology, Engineering, and Mathematics (STEM) content through experiential, hands-on and student-directed projects is likely to lead to greater achievement [1]. It is very important to develop an inexpensive framework that allows students to easily test their algorithms on a low-cost, real-time experimental test bed to support project based learning for college students.
Music beat tracking raises concerns in many researches recently [2]. In [3], the authors use high frequency content (HFC) and spectral difference (SD) to detect music beat. In [4], music beat tracking algorithm is used for heart sound segmentation and s1/s2 detection.

However, these algorithms only run on computers and hinder the experiments on mobile platform. In this paper, we present a low-cost music beat tracking system using Arduino. The rest of the paper is structured as follows. Firstly, the main electronic hardware components in the real-time music beat tracking system will be explained. Secondly, we discuss the workflow and information flow of the beat tracking system. Thirdly, the algorithms driving the real-time music beat tracking system will be described. Finally, the conclusion and ongoing work are presented.

BEAT TRACKING WITH ARDUINO

In this part, we describe the overall design of the real-time music beat tracking system. The pricing of the main components of the beat tracking system will also be discussed.

System Design

We use Arduino boards (ArduinoUno[4]) to read sensor inputs and turn it into an output – blinking a LED, for example. We can tell Arduino what to do by sending a set of instructions to the microcontroller on the board. To do so we use the Arduino programming language and the Arduino Software (IDE). The overall design of the real-time music beat tracking system is illustrated in Fig.1 below.

![Figure 1. The Overall Design of the Beat Tracking System.](image-url)
System Pricing

Table I lists the components of the real-time music beat tracking system and the cost for each at the time the paper was written. The hyperlinks of these components vendors on Taobao.com are also provided in Source column (note: the price may change over time). The total price of the music beat tracking system hardware is less than $5, and is affordable to every student.

<table>
<thead>
<tr>
<th>Component</th>
<th>Price(¥)</th>
<th>Price($)</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arduino Uno Compatible (with USB cable)</td>
<td>11.7</td>
<td>1.80</td>
<td>Taobao.com</td>
</tr>
<tr>
<td>KY-037 Sound Sensor</td>
<td>0.9</td>
<td>0.14</td>
<td>Taobao.com</td>
</tr>
<tr>
<td>Arduino Components Package (LEDs, Breadboard, Resistors, Wires)</td>
<td>9.9</td>
<td>1.52</td>
<td>Taobao.com</td>
</tr>
<tr>
<td><strong>Total Price</strong></td>
<td><strong>¥22.5</strong></td>
<td><strong>$3.46</strong></td>
<td></td>
</tr>
</tbody>
</table>

SYSTEM WORKFLOW

In this section, we discuss in detail about the workflow in the real-time music beat tracking system. Fig. 2 describes the main workflow and information flow in the platform. The Arduino board reads in the sound signals from KY-037 sensor attached on #1 analog pin, and compares the read in value with a predefined threshold value $T$. If the read in value is above $T$, the current time is recorded with a recurrent time array. Pair of candidate time intervals are compared to form the music beat. The flag LED is switched every time a beat is generated. The beat LED starts to blink accordingly to the recognized beat. A live video demonstration for a Chinese song is available at: https://luckh2.github.io/aiclass/online/xxdd.html

Figure. 2. Workflow of the Beat Tracking System.
ONBOARD ALGORITHM

The music beat tracking system’s onboard algorithm is illustrated in Algorithm 1.

Algorithm 1. Real-time Music Beat Detection Algorithm

Input: Sound signal $S$ ; Threshold Parameter $T$
Output: Flag signal $F$, Beat signal $B$

1. time ← 0
2. $t[3] ← \{0\}$ //recurrent time array to record 3 continuous time positions
3. pointer ← 0 //time array pointer
4. beat ← 0 //the recognized beat
5. $i1, i2 ← 0$ //candidate beat interval
6. While True Do
   7. time ← millis() //time Now
   8. Val ← $S$
   9. If (Val > T) Then
      10. $t[pointer++] ← time$
      11. If (pointer > 2) Then
      12. pointer ← 0
      13. If ($t[0] > 0$ && $t[1] > 0$ && $t[2] > 0$)
      14. $i1 ← t[(pointer+1)\%3] - t[pointer]$
      15. $i2 ← t[(pointer+2)\%3] - t[(pointer+1)\%3]$
      16. If (abs($i1-i2) < 50$ && abs($i1-i2) > 0$) // 0 ~ 0.05s
         17. $F ← \neg F$ // swith flag LED
      18. beat ← ($i1+i2)/2$
      19. If (beat > 200 && beat < 2000) // candidate beat ranges from 0.2 to 2s
      20. $B ← \neg B$ // switch beat LED
      21. delay(beat) // last for “beat” time long

CONCLUSIONS

In this paper, we describe a low-cost (less than $5) real-time music beat tracking system using open-source hardware Arduino. We are looking forward to more interesting projects using the beat tracking system, for example, combing the beat tracking system with autonomous robot and making a robot dancing to the beats of music. The beat tracking system is open sourced and the code for the project is available online. (https://github.com/luckh2/aiclass/blob/master/startup/beat.ino)

REFERENCES


211