Rhythm-Based Video Game Assessment

Maciej Draus, Lorenzo J. Tardón, Sergio Guillén, Ana M. Barbancho, Isabel Barbancho
macdraus@uma.es, ltg@uma.es, sergiogj@ic.uma.es, abp@uma.es, ibp@uma.es
ATIC Research Group, ETSI Telecomunicación, Universidad de Málaga, 29071 Málaga, Spain

***Abstract*—Humans have shown a natural tendency to move or adapt, intentionally or unintentionally, to the beat of rhythmic auditory stimuli, such as music. This activity is sustained by a complex neuronal network, including perceptual regions, motor regions and sensorimotor integration areas. These abilities can be trained by practising, in this context music-based video games are a great tool to improve those rhythmic skills, like hand-eye coordination or synchronization. An important aspect of this tool is the feedback the players get after playing, so they know what they do right and what they do wrong. Thus, feedback plays a leading role in player's improvement. The aim of this article is to develop an assessment scheme for a rhythm-based video game to help the users improve their rhythmic skills by playing. This scheme shows information like hit percentage, accuracy or the tendency to make early or late hits. In this way, only a few values provide users with a lot of feedback.**

I. Introduction

Rhythmic skills are natural among the general population; they refer to the ability to recognize and perform patterns [1]. These activities are sustained by a complex neuronal network in the human brain [2], and they have an important role in various areas, including music, sports or any activity where timing and hand-eye coordination are vital. Developing rhythmic skills has numerous benefits for physical coordination, cognitive function and creativity [3].

Video games can be an effective and, at the same time, fun way to improve rhythmic skills [4]. There are several types of rhythm-based video games that require players to follow a rhythm, keep a beat or perform rhythmic patterns. Assessment in these games usually measures player's accuracy and timing in performing the required actions with the rhythm. Then, the scoring system assigns points based on player's precision [5]. This scheme typically rewards players for being precise on the beat, and discounts points for missing notes. Within this context, in this manuscript we unveil the assessment scheme employed in 'Hammersong' video game.

This manuscript is organized as follows: in Section II, the game, and its goal are described; Section III exposes how data are treated for the evaluation; in Section IV, the scoring system is explained; and in Section V, it is shown how the players' performance is evaluated.

II. Game Description

'Hammersong' is a rhythm-based video game where player's goal is to perform a rhythmic pattern as accurately as possible, described by making use of Western music notation. In addition, the forge's Greek god Hephaestus is playing a rhythm too: at the easiest levels Hephaestus and the player perform the same rhythm; however, as the difficulty increases, the patterns become different from each other, thus adding another level of complexity (see Fig. 1).

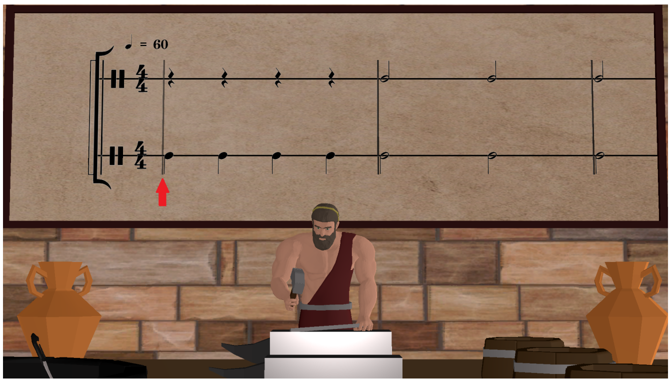

Fig. 1. Hammersong game snapshot.

Fig. 1 shows a level screen from 'Hammersong'. The board displays two rhythm patterns: the upper pattern corresponds to the player's rhythm and the lower one to Hephaestus rhythm. Both patterns are moving left at the same speed and players have to hit the screen when an event reaches the vertical reference bar marked with a red arrow (reference bar remains static).

The game generates data according to player's inputs and gives a score feedback when the level ends. These data give us the exact time the player hits each note, so it's not necessary to use any Onset Detection Functions (ODF), unlike in other schemes [6].

III. Data Analysis

The data gathered by playing the game are pre-processed, and analysed in order to evaluate the players' performance. The data generated by playing, and the data adaptation stages are described next.

A. Generated Data

There are four pieces of information stored for each input:

- Actual Time (*AT*): The precise time the player should play the note.
- Note duration (*ND*): The duration expressed as the number of sixteenth notes it is equivalent to.
- Note Type (*NT*): The type value is '0' for notes and '1' for rests.
- Measured Time (*MT*): The instant when the player actually played the note.

These elements of information are graphically shown in Fig. 2.

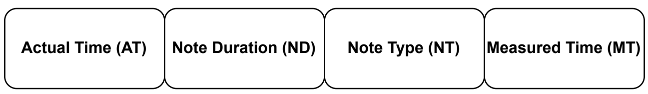

Fig. 2. Generated data structure for each input.

B. Special cases

In some cases it is necessary to overwrite the initially obtained MT to represent the errors correctly. Specifically, the following cases were identified:

1) Not Played Note or Played Rest: They are considered as maximum errors. The Maximum Error, (ME), is defined as half the note's duration, and it is calculated for each note using the beats-per-minute values, BPM, and note's duration using eq. (1).

$$ME = (ND * 60)/(8 * BPM) \tag{1}$$

Finally the new MT is calculated with eq. (2).

$$MT = AT + ME \tag{2}$$

2) Not Played Rest: This case is considered as a perfect hit. So the new MT is equal to the AT (eq. (3)).

$$MT = AT \tag{3}$$

3) Several inputs for the same note: In case the player plays within a note's interval more than once, the MT, for multiple inputs, is set to the maximum error (eq. (2)).

C. Delay

The player's delay, Δ, at each event is obtained from the generated data by using eq. (4).

$$\Delta = MT - AT \tag{4}$$

In this case, positive delay represents playing a note too late, and negative delay otherwise.

D. Data Representation

Fig. 3 represents the delay signal obtained in the previous step. The black horizontal line shows the AT to make it easier to differentiate between early and late hits. Each individual input is displayed as a red asterisk.

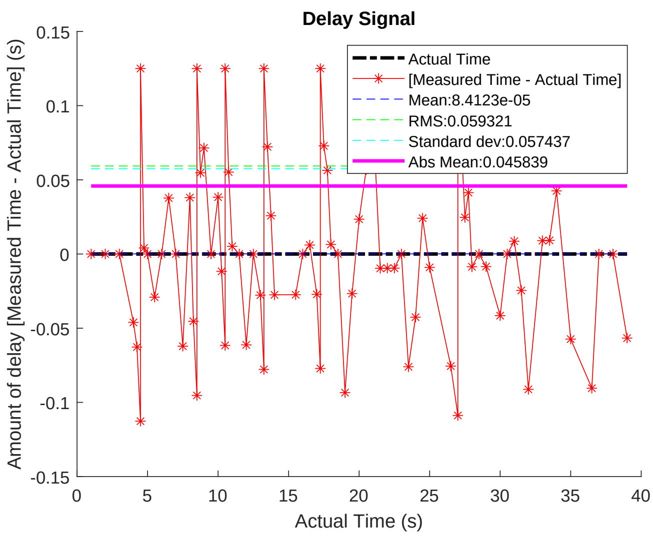

Fig. 3. Delay signal representation.

Figure 4 represents the error the player commits with respect to ME across time. This percentage is obtained by normalizing the delay to obtain the Relative Error (RE) as given by eq. (5):

$$RE = \frac{|\Delta|}{ME} \tag{5}$$

Finally, RE is multiplied by 100 to get the percentage of error.

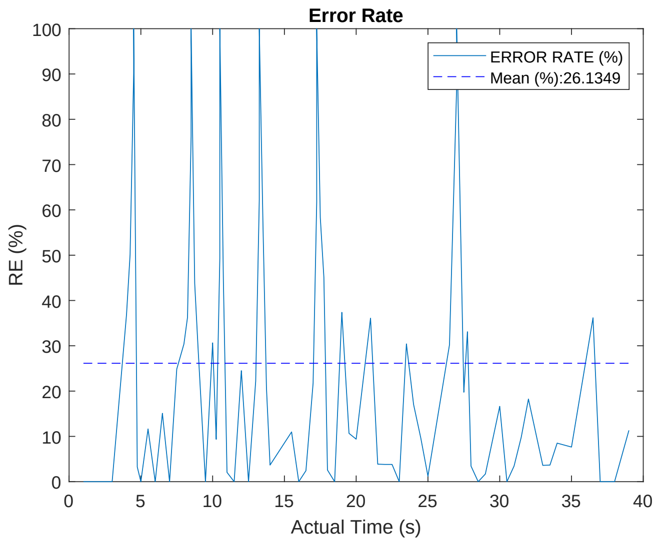

Fig. 4. Illustration of relative error (RE) across time.

IV. Scoring System

After the data for rhythmic evaluation are gathered and pre-processed, scores can be calculated for the players' performance. The procedure followed is described next.

A. Right and Wrong Notes

The persistence of hearing is a concept related to the fact that the sensation of hearing any sound persists in our brain a certain time. The human ear can distinguish two consecutive sounds if they are at least 70 ms apart for dry sounds and 100 ms apart for complex ones [7]. The 70 ms threshold is the tolerance (T) considered to separate hits from misses:

$$T = \pm 0.07 \text{ (s)} \tag{6}$$

Thus, player's inputs are divided into four groups depending on the delay (see Fig. 5):

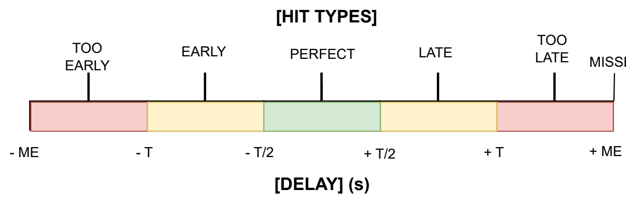

Fig. 5. Hit types according to delay.

1) Perfect Hits: Perfect hits represent the highest score the player can achieve for a single note. A hit is considered perfect when eq. (7) is met:

$$-\frac{T}{2} \text{ (s)} < \Delta < \frac{T}{2} \text{ (s)} \tag{7}$$

Note that in this way, a window of 70 ms of duration is defined for perfect hits.

2) Normal (Early and Late) Hits: Normal hits get lower score than perfect ones, but they are still considered as correct. They are Early when eq. (8) is fulfilled:

$$-T\ (\text{s}) < \Delta < -\frac{T}{2}\ (\text{s}) \tag{8}$$

Or Late if eq. (9) is met:

$$\frac{T}{2}\ (\text{s}) < \Delta < T\ (\text{s}) \tag{9}$$

3) Failed (Too Early and Too Late) Hits: Failed hits decrease the player's current score. They are Too Early when eq. (10) is fulfilled:

$$-ME\ (\text{s}) < \Delta < -T\ (\text{s}) \tag{10}$$

Or Too Late if eq. (11) is met:

$$T\ (\text{s}) < \Delta < ME\ (\text{s}) \tag{11}$$

4) Missed Hits: If the player does not hit an event or he/she hits a rest, a missed hit is considered, and eq. (12) is used to define the delay:

$$\Delta = +ME\ (\text{s}) \tag{12}$$

V. Evaluation

Once the player's inputs have been evaluated and classified according to their delay, some additional considerations are taken into account to provide information to the user about their performance.

A. Grade

A weighted mean is employed to decide the player's grade. Table I shows the weights used for this calculation.

TABLE I
GRADE WEIGHTS

Hit Types	Weight
Perfect (*P*)	+2
Normal (*N*)	+1
Failed (*F*)	-1
Missed (*M*)	-2

Consequently, the Average Grade ($\bar{G}$) is defined by eq. (13):

$$\bar{G} = \frac{2P + N - F - 2M}{P + N + F + M} \tag{13}$$

where P, N, F, and M represent the number of perfect, normal, failed, and missed events, respectively.

Next, the value obtained from eq. (13) is used to determine the player's grade depending on the range in which it falls. Grade ranges are drawn in Fig. 6. The final grade is represented by a letter, and it's displayed to the player every time a level is completed: it represents the overall goodness of the player's performance [6].

B. Score

An easily comprehensible Score (S) is also obtained from $\bar{G}$ by readjusting the range from $[-2, 2]$ to $[0, 10]$ using eq. (14):

$$S = 2.5(\bar{G} + 2) \tag{14}$$

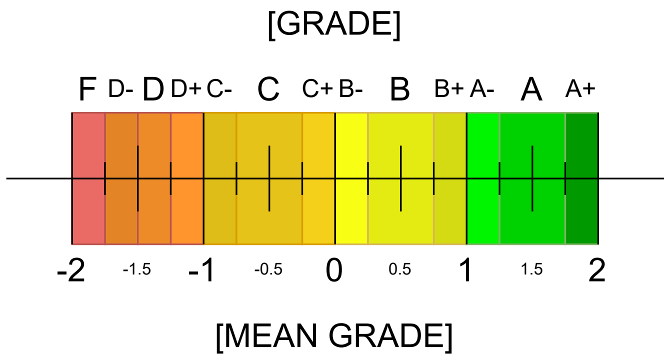

Fig. 6. Diagram of Average Grade intervals.

C. Tendency

With the grade, the player has no information on whether his/her hits are early or late. For this reason, the Tendency (or bias), B, is defined and shown to inform the player what type of deviation he/she is most prone to commit.

Again, a weighted average is calculated using other hit types and weights shown in Table II.

TABLE II
TENDENCY WEIGHTS

Hit Types	Weight
Too Late (*TL*)	+2
Late (*L*)	+1
Perfect (*P*)	0
Early (*E*)	-1
Too Early (*TE*)	-2

The Average Tendency ($\bar{B}$) is obtained from eq. (15):

$$\bar{B} = \frac{2TL + L - E - 2TE}{TL + L + P + E + TE} \tag{15}$$

where TL, L, P, E, and TE represent the number of too late, late, perfect, early, and too early events, respectively.

The overall tendency assessment is selected according to the range in which the Average Tendency lays, as it is displayed at Fig. 7. The Tendency messages can be: Too Early (TE), Early (E), Balanced (BA), Late (L) and Too Late (TL).

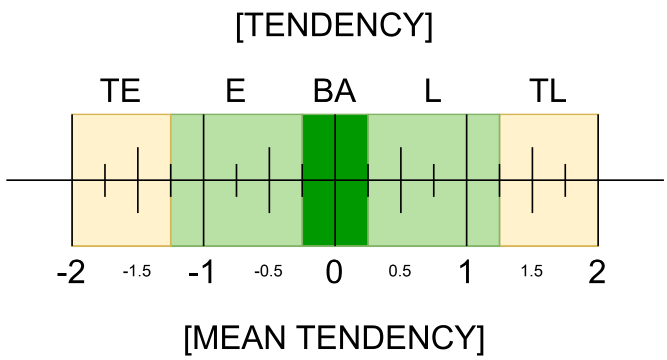

Fig. 7. Diagram of Average Tendency intervals.

D. Feedback

Finally, the values measured, and corresponding messages are displayed on the screen to show the players their performance assessment at the end of each level. The evaluation menu is shown in Fig. 8.

Fig. 8. Evaluation feedback shown to the player.

VI. Conclusions

Training for the improvement of rhythmic skills has great impact, and benefits motor coordination, cognitive functions and creativity.

Rhythm-based video games are a tool to train and improve the player's skills in an entertaining way.

This task has been considered and carried out by developing a rhythm-based game, and an evaluation scheme.

The feedback displayed at the end of each level is an important part for skill improvement; the specific choices and developments to provide the players with readable feedback have been exposed. The evaluation is aimed at letting the player know how good his/her performance was, and what he or she did wrong; thus, facilitating the correction of errors on next games, and the improvement of rhythmic perception and coordination in the musical context built in the game. Also, the performance of the player allows him/her to progress across multiple levels.

Some aspects like the addition of background music, the inclusion of known rhythmic patterns or the design of rewarding schemes are aspects that can be included to improve the game and attractiveness.

Acknowledgements

This publication is part of the project PDC2021-120997-C33 funded by MCIN/AEI/10.13039/501100011033, and European Union "NextGeneration EU/PRTR". This publication is part of the project PID2021-123207NB-I00 funded by MCIN/AEI/10.13039/501100011033/FEDER, UE. This work was done at Universidad de Málaga, Campus de Excelencia Internacional Andalucia Tech.

Special thanks to Francisco Vargas Boza (Programmer), Juan Carlos Camuña Cotta (3D Modeler) and Isabel Tardón (2D Character designer) for their contribution to the development of the video game 'Hammersong'.

References

[1] Bruno H Repp and Yi-Huang Su. Sensorimotor synchronization: a review of recent research (2006–2012). *Psychonomic bulletin & review*, 20:403–452, 2013.

[2] Valentin Bégel, Ines Di Loreto, Antoine Seilles, and Simone Dalla Bella. Music games: potential application and considerations for rhythmic training. *Frontiers in human neuroscience*, 11:273, 2017.

[3] Amanda Pasinski. Possible benefits of playing music video games. 2014.

[4] Gianna G Cassidy and Anna MJM Paisley. Music-games: A case study of their impact. *Research Studies in Music Education*, 35(1):119–138, 2013.

[5] Dmitry Alexandrovsky, Katya Alvarez, Benjamin Walther-Franks, Joschua Wollersheim, and Rainer Malaka. Jump 'n'rhythm: a video game for training timing skills. *Mensch und Computer 2016–Workshopband*, 2016.

[6] Başak Esin Köktürk-Güzel, Osman Büyük, Barış Bozkurt, and Ozan Baysal. Automatic assessment of student rhythmic pattern imitation performances. *Digital Signal Processing*, 133:103880, 2023.

[7] Robert C Wingfield. An experimental study of the apparent persistence of auditory sensations. *The Journal of General Psychology*, 14(1):136–157, 1936.