Kinect vs. Color Gloves: A Music Game tool for Hand Tracking Evaluation

Lucas Silva Figueiredo and Veronica Teichrieb
Voxar Labs - Informatics Center of the Federal University of Pernambuco

Abstract

This paper details the development and use of a gesture based interaction game as an entertainment system but, also as a hand tracking evaluation tool. By using the ”air guitar” concept allied to music games interfaces, the proposed work performs an analysis of two different hand input methods. Firstly, a color glove detection method is used, then the Microsoft Kinect sensor is integrated providing an attachment-free tracking method. Tests were performed and a set of users evaluated both the game and the tracking methods. The results showed the impact of each tracking method on the user playing experience as well as the users understanding of the comparison game tool, validating the use of the proposed music game for the task.

Keywords:: Music Games, Hand Tracking Evaluation

Author's Contact:

{lsf,vt}@cin.ufpe.br

1 Introduction

The concept of music games refers to video games in which the gameplay is meaningful and often almost entirely oriented towards the player's interactions with a musical score in individual songs. In the last decade, they have conquered a significant space on game market. On the other hand, recently, Microsoft presented the Kinect sensor [Microsoft 2011], a device containing depth-sensors, camera and microphone, which is capable of replacing conventional controllers. Such technology would allow a person to use his/her own arms and legs to directly interact with the game [Shotton et al. 2011]. Moreover, this controller-less approach may be applied to games similar to Rock Band, by using the idea of playing an air guitar, which refers to imitate rock guitar gestures along the music without the actual physical instrument [Crane 2006]. This practice is more showmanship than musical performance and does not require real musical skills. Generally, the practice of ”air guitar” is much more related to a theatrical performance than to a way of interaction. However, due to its metaphorical and corporal expression essence, the concept of playing an imaginary guitar can be used as a powerful gestural interaction tool for music games.

In the last few years some initiatives were developed in this direction. In [Mäki-Patola et al. 2005] a virtual guitar is proposed, being controlled only by user hand movements, and providing a sound output whenever it is ”played”. Furthermore, in [Figueiredo et al. 2009] an open-source library is published, supplying incoming applications with a highly customizable gesture interface for an ”air guitar” performance, with several options of sound output, including a tone mapping tool, a slide effect simulation and a playback system in order to backup the user solo performance with a sonorous environment. Those examples showed to be successful as an interaction tool, revealing, in most cases, the user satisfaction and joy. However, both simulators targeted the virtual guitar much more as a musical expression interface, leaving unexplored its facet as a game controller. In both cases the hand positions were tracked through the use of colored gloves and image processing algorithms.

Beyond the game itself as an entertainment application, the current paper aims also to present it as a hand tracking evaluation tool. Due to music games mechanics, which requires the player to perform the specific commands within a considerably short time and space range, it is possible to analyze different input methods in a compatible way. Therefore, all relevant game statistics derived from user interaction are stored and displayed at the end of each game session, in order to perform a comparative study. Furthermore, two different interaction methods were incorporated into the game prototype to be evaluated: the first consists of a color tracking algorithm, set to detect two orange gloves as suggested in [Mäki-Patola et al. 2005] and [Figueiredo et al. 2009]; the second one uses the body tracking provided by the Kinect in order to obtain the hand positions information. Thus, the executed tests provided useful information for the analysis and comparison of these two tracking methods in the context of music games.

The paper is organized as follows: in Section 2 some of the main related works are cited; Section 3 defines the main concept of the game and its evolution through some interface scenarios; Section 4 approaches the two tracking methods used; Section 5 is dedicated to the case study, the obtained results and related discussions; finally, Section 6 contains the conclusion of the presented work.

2 Related Work

Many works have their piece of contribution to the idea introduced in this paper. Among them, music games play their role as one of the main motivator of this work, as well as a source for the study of interfaces and player interaction. Games like [Games 2007] and [Ins 2010] contributed with ideas of interaction and interface presentation, as well as the air guitar simulators presented in [Mäki-Patola et al. 2005], [Pakarinen et al. 2008] and [Figueiredo et al. 2009]. Moreover, the recently launched Kinect showed to be an useful tool for body tracking purposes, being also are considered in addition to the color glove tracking used in the aforementioned virtual guitars simulators. The Kinect nowadays SDKs (OpenNI [Sense 2010] and Microsoft Kinect SDK [Microsoft 2011]) are able to through a trained system, track peoples 3D skeletons by using only the input depth image of the sensor [Shotton et al. 2011].

3 Game Mechanics

The proposed game is a music game based on spatial gesture interaction. Likely Rock Band, the objective in this game is to execute the indicated moves by incoming arrows, continuously, meanwhile a song is played. The execution should occur synced with the sonorous elements of the song, being the execution of the current move happening at the same moment in which determined sound became evident. This way, the user can make a direct association of the performed move with the played song. In addition, we propose that the user must experience a more freely interaction mode, reusing the idea of air guitar as an interaction tool.

The presented game is built upon the open-source framework provided by [Figueiredo et al. 2009], in which a virtual guitar is simulated by tracking only the player hands. On the following text, in order to simplify the proposal, it will be supposed that the player left hand is the hand referred to the virtual guitar arm, and the right hand is the one over the guitar body.

3.1 Spatial Gestures

Beyond simulating a virtual guitar, in [Figueiredo et al. 2009] a method of interaction is described that is based on the recognition of two simple movements. The first movement is related to the vertical swing of the right hand. This movement represents the user intention to play the imaginary guitar strings, being analogous move to a real guitarist when it is desired to play the six guitar strings all at once. The second movement is related to the horizontal translation of user's left hand (which is over the guitar arm), and it is analogous to the guitarists moves when it is desired to change from low to high (and vice-versa) pitch levels while playing notes or chords.

In order to improve the game interaction, making it more dynamic as well as more coherent to the proposed metaphor, both directions of the right hand vertical movement were considered: from up to down and vice-versa. As an indication of which direction the move should be performed, each command is expressed by an arrow pointing down or up, corresponding to the demanded right hand movement.

The right time to perform the move is indicated by the base line of the platform in perspective (which is the support plane whereby the arrows come down), at the bottom part of the screen, just before the arrow disappear from the display. This way, the player knows in advance the next command to be executed and can prepare the right hand swing direction, being able to successfully perform it within a time range of approximately 200 ms. This and other concepts can be perceived in the game interface, illustrated in figure 1.

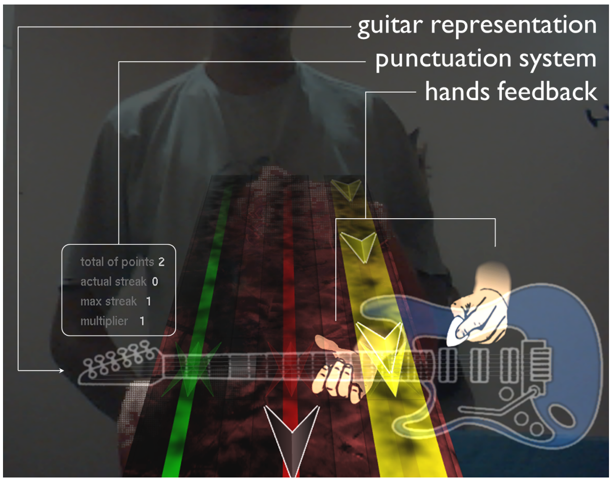

Figure 1: *Game interface, showing the arrows commands, colored rails, guitar and hands representation in a unified way*

The number of rails can be configured, however during the development it was empirically verified that this number must not be too large. Indeed, requesting the user to change between five rails is already a tricky quest. It occurs mainly due to the virtual guitar nature of being a floating device, not fixed and not physically represented, making it difficult to the user change between rails precisely. Furthermore, the more rails are used, more refined must be the user's skill, and since the objective is to build a wide range usable game, it is not interesting to introduce users to the proposed game with such difficulty level. This way, all formal tests were realized using a three-rails virtual guitar.

3.2 Performance Measurement

In order to measure the player performance, as well as to give the player stimuli to achieve his best in the game session, a score system was developed. Following the models presented in Rock Band, three main measures were tracked over time.

The first one is the number of the consecutively correctly hit arrows, storing and incrementing a counter whenever the player makes a right move, and bringing it to zero whenever a failure occurs. At the end of the song session, the higher counter value obtained through time is registered. Another important measure is the percentage of hit arrows over the failed ones. Moreover, for each arrow it is stored if it was hit or missed and if missed, for what reason, which could be one or a combination of the following: the left hand was not occupying the correct track at the moment; the right hand crossed in the wrong direction; or the right hand did not cross over the virtual guitar strings. This way, it is possible to know not only if the player went well on his execution, but also for what reasons and in which intensities he lacked performance. Besides that, whenever the right hand crosses and there is a current arrow to be played, no matter if the user hit or missed it, it is stored the time gap between the movement and the perfect middle position that an arrow was (or would be in an instant) in the playable zone. This last data helps to evaluate the impact of delay on the players performance.

4 Hand Tracking Methods

Two hand tracking methods were chosen and further integrated to serve as input source for the proposed music game.

The first hand tracking method used is the one proposed in [Figueiredo et al. 2009], which requires the users to wear gloves with contrasting colors in relation to the ambient in order to further apply a segmentation process, isolating the user hands from the remaining parts of the scene. The role of the glove tracking algorithm is to find the two groups of pixels that represent both player's hands. This information is used on the estimation of hands' screen position. Due to the previous experience of this method in an air guitar application, which showed to be successful after usability tests, the color glove detection method showed to be the straightforward option to be used on the proposed game.

Secondly, the Kinect Sensor was used as input. The Kinect based detection and tracking of the human body is made based on training, where a skeleton model is attached to the depth image captured [Shotton et al. 2011]. This way, the sensor was used as a hand tracking tool by selecting the retrieved hands points of the skeleton as an estimation of hands' screen position likewise the glove hand tracking method.

5 Case Study and Results

In order to evaluate the proposed game, as well as its attached interaction methods a case study was prepared. Firstly, by using an composer module, a list of commands were generated for the selected song, Paint it Black by Rolling Stones. This song has some characteristics which turns to be important as a first experience in a music game. It has a pattern that is repeated in a way which permits the listener to predict where it goes. It also has a rhythmic and not fast dynamic giving the players time enough to change positions between the incoming commands. The environment was then prepared for the users, to provide some free space as well as a not far visualization of the game display, as shown in figure 2.

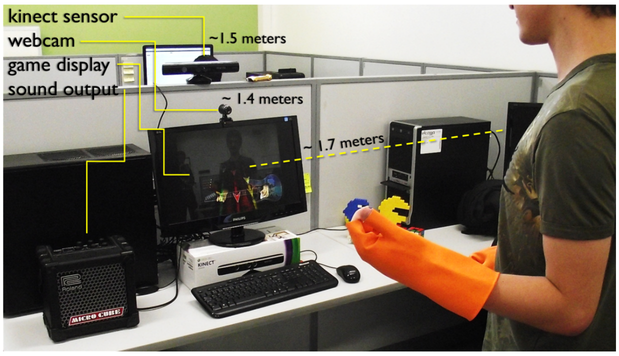

Figure 2: *Case study environment*

Twelve subjects participated in the tests, 6 men and 6 women with the age varying from 18 to 26 years old. Each user played the same song with the two tracking methods, varying their order equally among the users. Firstly the users were presented to the system, with a brief explanation and then they where put to play the song for 2 minutes, in order to give them some initial experience so they could gain more familiarity with the system and feel more comfortable with the following evaluation step. Then, they were invited to perform their best score in the game, during the given song until its last command, totalizing a time of 2 minutes and 21 seconds of game session. After the play session, each tested subject responded to a survey about the experience. The questions were mostly based on [Pinelle et al. 2008], which reveals twelve main issues commonly present in video games of all types. From these, a subset of five (which were more related to the interaction itself), fully matched with the game proposal.

1. Unpredictable response to user's actions: poor hit detection; inconsistent response to input.
2. Difficult to control actions in the game: oversensitive controls; unnatural controls.
3. Response to user's action not timely enough: slow response time interferes; old commands overloading the current performed.
4. Exaggerated complex command sequences: learning curve is too steep; sequences are complex, lengthy, and awkward, making the game difficult to play.
5. Visual representations are difficult to interpret: bad visualization of information; too much screen clutter; too much elements on the screen at the same time; difficult to visualize and distinguish interactive content from the non-interactive content.

To quantify the users results, a Likert scale was used. For each question the user should answer in a scale from 1 to 5, from totally disagreed to totally agreed with the affirmative. The first six questions were directed explicitly to the proposed game interface itself and the played music as well. The following six questions aimed to evaluate the tracking methods. All questions are presented bellow:

1. In general, it is easy to interpret the visual representations provided by the game.
2. The game presents a clear interface, free of noise.
3. The number of visual elements displayed simultaneously on the screen is ideal.
4. In the game screen, it is easy to distinguish the interactive content of the non-interactive.
5. In general, the sequences of commands are easy to perform.
6. The learning curve is easy, adapting and executing the task presented evolves with very little time of use.
7. The response to the player's actions is consistent.
8. Every time a command is sent by the player intentionally, it is detected.
9. The game does not see inexistent movements; only computes commands actually made by the player.
10. It is easy to control the actions in game.
11. The control sensitivity is correctly set.
12. The responses to the commands sent to the game are provided instantly.

The figure 3 details the results by showing each question punctuation as a sum of users ratings. Since the number of users is twelve, and the maximum of points to give for each question is five, the maximum score is 60, an intermediate score is nearby 36 and a low score is between 24 and 12. It can be observed that in relation to the game implementation itself, considering its presented interface, and the tested song commands, the overall score showed to be highly satisfactory (all presents scores are between 50 and 56). This is an indicative that the game, by its turn, showed not to be a setback on players performance in relation to the user understanding of the proposed task and goals. The presented difficulty of the required sequence of commands showed to be an adequate task regarding its level of difficulty. This way, this indicative establishes trust in a way that the final results main issues are more directly impacted by the tracking methods than by the understanding of the game.

Moreover, the users also evaluated the tracking methods, demonstrating that the color gloves tracking showed the best satisfaction results. Regarding the tracking methods overview, it is noticed that in most topics, the Kinect achieved an intermediate score, revealing deficiencies related to the coherence, false-positives and false-negatives, which suggests that the tracking of Kinect lacks in precision, and so may present some jitter problems, causing the users to feel less in control over the game, impacting their performance as a whole. Furthermore, the users also evaluated the response time in question twelve and it is noticed that the Kinect presents a little delay which may be caused by the tracking algorithm processing time or a consequence of a drift effect, which occurs in case the tracking makes use of previous frame information to estimate the current position of the target. The color gloves, by its turn, showed to be a very successful tracking method for the proposed task. All questions scores regarding the color gloves achieved values between 53 and 58, revealing a great satisfaction by users. In figure 4 it can be seen that mainly the question 9 and 11, which ask about false-positives and control sensitivity respectively, had ratings 2 related to the Kinect; this rating does not occur a single time with the color gloves ratings. In general, as can be seen in figure 4, the ratings about the Kinect Sensor are grouped near the rating of 4, and in concerning of the color glove tracking method, the ratings are grouped near 5.

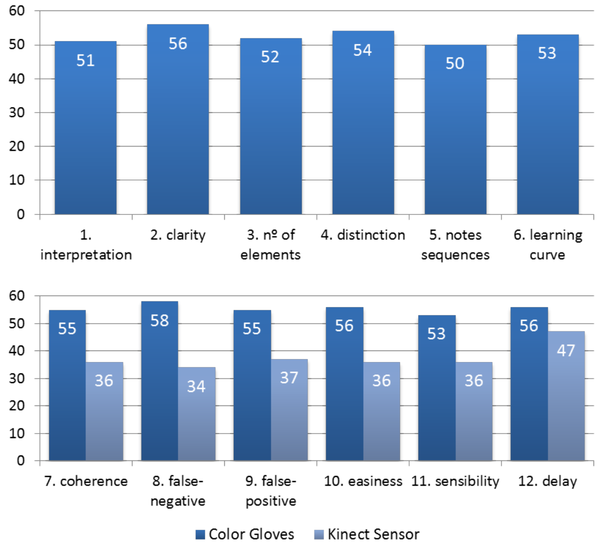

Figure 3: *Top: users feedback about the game interface and dynamics. Bottom: overall results about the used tracking methods*

Table 1: *Comparison of the results of both tracking methods through game measured values*

input method	% of hit	max streak	sleep notes	wrong rail	delay
Color Gloves	80.2	38.1	24.583	6.417	7,23
Kinect Sensor	68.8	29.5	35.333	11.917	14,27

The table 1 shows the average values of the gathered data. The percentage of hit notes (arrows) denotes the part of the required commands which the users performed correctly. The percentage is an important measure to understand users performance more generally, analyzing how much of the task was executed correctly. It also represents the overall performance of the hand tracker. Any issues regarding latency, precision, and tracking failures would impact in this result. The max streak by its turn is more related to the tracking stability, since a tracking method which eventually fails, or lacks precision, makes impossible the task of the user to keep hitting notes consecutively. The sleep notes and the wrong rail columns specify the cause of error regarding the users non-hit notes (considering that the amount of notes in the session is 157). The sleep notes measure is an important indicative of time response issues, since if the tracking is slow or presents drift problems, the user may miss some notes even if the movement was performed correctly. The wrong rail error is an indicative of precision problems. It means that the user missed the incoming notes not because he did not perform the right hand movement, but because he performed it with the left hand placed over a wrong rail. This way, this case of error denotes a possible failure of precision by the tracking technique, e.g. a jitter problem may cause the system to move against the user will and place his virtual left hand over the wrong rail. Finally, the delay represents the average of all stored time gaps between the ideal moment the user should have played a note , and the moment he actually played it. This measure indicates also the response time of the used method and may explain for example, the sleep notes value.

By analyzing the table 1, it is possible to notice that the users per-

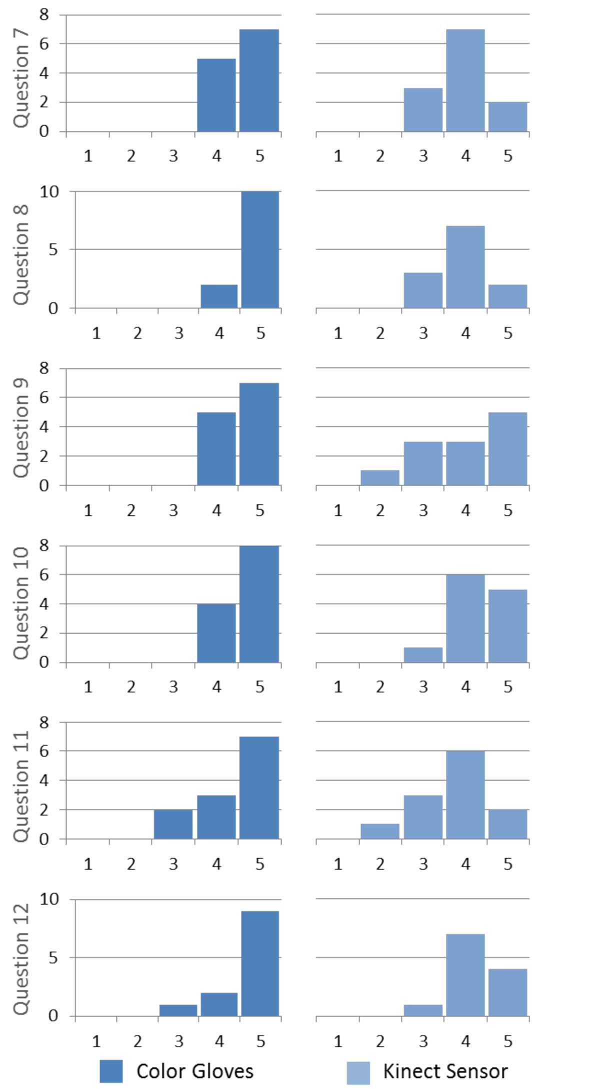

Figure 4: *Results of users ratings about each comparative question directed to the tested tracking methods*

formance was significantly better with the color gloves, than with the Kinect sensor. The first important thing to be noticed is that the results presented by the gathered data during the game sessions match the users feedback in the survey results. This means that as an evaluation tool, the proposed music game showed to be coherent regarding the users experience. And so, the presented tool shows to be capable of correlating the hand tracking method performance, in a real scenario application, with the users impression of the control method used. This way, the proposed game besides providing a report of the hand tracking methods tests, shows results regarding the tracking real use, attempting to give a meaningful evaluation to users and developers which intend to use a specific tracking algorithm or device.

Moreover, as discussed before, the results suggest that the Kinect tracking lacks of precision causing the users to miss notes along the sessions. Another indicative is that the Kinect presents a greater delay and so its response time is affected. The consequence of this latency issues is explained by the higher number of sleep notes the users had with the Kinect sensor. Indeed, the processing time of each tracking method was considerably low, and both of them filled the real-time constraint requisite. The Kinect processing time is about 5 ms and the color glove tracker hits 3 ms. Nevertheless the low processing time, the Kinect presented response time issues, meaning that the input delay promoted by the Kinect is due to its tracking method which incorporates a little of the drift effect. The glove detection does not suffer from this problem due to its nature of using only the current frame information only to perform the tracking, in other words, a tracking by detection method. Finally, the full processing time of the game varies from 14 to 18 ms. All tests were executed on a Intel(R) Core(TM) i5-2300 CPU @ 2.80 GHz, with 4.00 GB RAM, running on the Windows 7 Enterprise x64 OS.

6 Conclusion and Future Work

The presented work showed a gesture based music game which can be played using as input the tracked positions of the user's hand. The game is also presented as a tool for evaluation of hand tracking methods. Both the game and two different input techniques were evaluated, revealing satisfactory results related to the proposed interface and gesture interaction. In more detail, from the two tested methods, the color glove detection showed to be the most successful, achieving the best results by the users evaluation and also by the numerical results itself. The second tracking method used was provided by the Microsoft Kinect device, which presented, in a comparative analysis, some issues regarding its response time and precision.

For future work, there are foreseen improvements in the game mechanics aiming to correlate more precisely the game gathered data to the issues presented in the used tracking methods. Futhermore, a set of focused songs is desired in order to investigate more specifically and separatedly each problem.

Acknowledgements

The authors gratefully acknowledge CNPq and FACEPE for the financial support of this work.

References

CRANE, D., 2006. To air is human: Oneman.s quest to become the world.s greatest air guitarist. Riverhead Trade.

FIGUEIREDO, L. S., TEIXEIRA, J. M. X. N., CAVALCANTI, A. S., TEICHRIEB, V., AND KELNER, J. 2009. An open-source framework for air guitar games. In *Proceedings of the 2009 VIII Brazilian Symposium on Games and Digital Entertainment*, IEEE Computer Society, Washington, DC, USA, SBGAMES '09, 74–82.

GAMES, M., 2007. Rock band. http://www.rockband.com/.

2010. Instant jam. http://web.appstorm.net/how-to/social-media-how-to/social-music-gaming-with-instant-jam/.

MÄKI-PATOLA, T., LAITINEN, J., KANERVA, A., AND TAKALA, T. 2005. Experiments with virtual reality instruments. In *Proceedings of the 2005 conference on New interfaces for musical expression*, National University of Singapore, Singapore, Singapore, NIME '05, 11–16.

MICROSOFT, 2011. Kinect for windows. http://kinectforwindows.org/resources/.

PAKARINEN, J., PUPUTTI, T., AND VÄLIMÄKI, V. 2008. Virtual slide guitar. *Comput. Music J. 32* (September), 42–54.

PINELLE, D., WONG, N., AND STACH, T. 2008. Heuristic evaluation for games: usability principles for video game design. In *Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, ACM, New York, NY, USA, CHI '08, 1453–1462.

SENSE, P., 2010. http://75.98.78.94/Documentation.aspx.

SHOTTON, J., FITZGIBBON, A., COOK, M., SHARP, T., FINOCCHIO, M., MOORE, R., KIPMAN, A., AND BLAKE, A. 2011. Real-Time Human Pose Recognition in Parts from Single Depth Images. *Computer Vision and Pattern Recognition* (June).