

Haptic Relay - Including Haptic Feedback in Online Video Game Streams

Tony Morelli^(✉)

Department of Computer Science, Central Michigan University, Mount Pleasant, USA
tony.morelli@cmich.edu

Abstract. The popularity of streaming video game sessions has been on a constant increase. Players stream their games as well as cameras facing them, while providing a description of the game play to the viewers. Fellow players watch the games to learn tips and tricks, as well as to learn what to expect when playing the game themselves. This is a great learning tool; however, the current rendition of streaming video games is missing a key factor in video game satisfaction - tactile feedback. All mainstream video game consoles contain some sort of haptic feedback in the form of a rumble and provides greater immersion in to the game; however, this modality is missing from the live streams. Haptic information in some games is also essential for game play by people who are visually impaired. This paper presents Haptic Relay, a method of providing real time haptic feedback for online video game broadcasts.

Keywords: Haptic interfaces · Games · Streaming media

1 Introduction

Streaming video games across the internet is becoming more popular; however, the vibration cues contained within all modern day console controllers and phones is not present in the online streams. The research presented here is an approach to broadcast haptic cues to better replicate the gameplay experience and to lower the learning curve of new games experienced by game players who are blind.

In 2010, the United States passed the Communications and Video Accessibility Act. This Act requires advanced communications and video broadcasting to be accessible to people who are deaf, blind and deaf-blind. Although the applicability of this Act to video game streams has not been questioned, video game systems are covered under this Act and in the future video game streams may be considered part of this as well. If that is the case, the ideas presented here may be useful for video streams to be compliant.

This paper is organized as follows. First, background information is given on the use of haptic cues in video games, the rise of internet video game streams, and the use of haptic feedback in video game research. Then an approach is described which could enhance online streams and haptic video game. Finally, future work and a conclusion are described.

2 History of Haptics in Games

Although vibration (or rumble as it was first known) is standard in modern day gaming systems, it has not always been present. The first mass release of a vibration unit to be used in video games was the Rumble Pack released for the Nintendo 64 in 1997 [1]. This add-on to the controller allowed developers to vibrate the controller when exciting things happened on the screen such as explosions, or when a car was traveling off the road. This innovation was released with the hopes that players would obtain a more realistic gameplay experience while using the device.

Just a few months after Nintendo released the Rumble Pack, Sony released the Dual Shock controller [2]. The Dual Shock controller was named because it contained a vibration motor located against the palm of each hand when holding the controller. This has become the standard location for vibration motors in current generation gaming systems such as the Xbox One and Dual Shock 4 (standard controller for the Playstation 4) which also contain this same design.

The use of haptic feedback in games has been involved in legal battles that have cost the major players in excess of \$100 million in legal penalties [5, 6]. Mainly fighting over patents owned by Immersion Corp [4] and Virtual Technologies [3], Sony and Microsoft have invested a lot of money to ensure haptic capabilities remain standard on their game consoles.

3 Streaming Video Games

Twitch has made streaming video games mainstream and popular. In fact, 1.8 % of internet traffic comes from Twitch live streams [8]. Behind Netflix, Google, and Apple, is Twitch. According to Business Insider [9], Twitch accounts for more than 43 % of all live streaming content on the internet and it is producing enough money that people are making a living live streaming games with some reporting more than \$100k in yearly income. Business Insider also reports Twitch has more than 55 million active users and 58 % of them have reported an increase in watching live video streams while decreasing the time spent watching videos from other sources including television, and other non-live streaming sites such as YouTube. Amazon purchased Twitch for just under \$1 billion [10].

YouTube's most popular broadcaster is PewDiePie, a person who reviews video games and receives over \$4 Million a year in revenue [11]. He has over 42 million subscribers to his channel [12]. Video game streaming is very popular and lucrative; however, the streams contain audio and visuals only, the haptic channel is absent. Any kind of enhancement to the video streams may increase the engagement time of viewers and could result in larger revenues for broadcasters and providers due to the potential of selling more advertising.

4 Haptics in Research

The use of haptic feedback in games has been shown to be important. Players prefer haptic feedback to audio when playing full body 3D interactive games [13]. In general, virtual environments have shown to have a higher level of quality of experience when compared to environments that are only audio and visual based [14]; however, this level of experience has not also increased the performance in games [15].

Haptic cues have also been used when creating games for people with disabilities. These games have been used in rehab [16, 17] where the results of the rehabilitation have been better when haptic feedback is used. Games using haptic cues have also been created for people who are blind [18–20]. It is very important for people who are blind to have this additional method of communication to replace required visual aspects of games that are not represented by sounds.

This paper defines methods of haptic capture and retransmission suitable for online game play session video streams. The following sections describe methods of haptic capture, haptic transmission, and haptic replay demonstrated by a small field test. Standards of encoding text with video are analyzed and modifications are suggested to also carry haptic information and finally limitations of the current prototype and future work are discussed.

5 Haptic Capture

Two methods of haptic capture are described in this section. Direct haptic capture involves embedding methods within the game software in order for the game itself to transmit the haptic information. Indirect haptic capture involves the use of a third party capture device that will be able to capture and transmit haptic information from haptic capable controllers.

Direct Haptic Capture. The direct haptic capture is available to games that have haptic relay capabilities embedded directly within the game's source code. These games are required to have full knowledge of the haptic relay protocol described below, and may often be custom games created to comply with this system. Software developers will have to modify existing game such that any code referencing a change in status of any of the haptic motors will also have to inform the remote server of the change in real time.

Indirect Haptic Capture. This method of haptic capture is used when access to the game's source code is not available. A hardware modification to the controller is performed such that the intensity of the vibration motors is recorded and transmitted to the helper service for web disbursement.

A prototype of this method was created using an Xbox 360 wired controller attached to a PC running Windows 8. The Xbox 360 controller contains two different vibration motors located near the bottom of each side of the controller (Fig. 2). The intensity of these motors can be individually controlled through software and is electronically controlled by varying the voltage presented to each of the motors. The higher the voltage, the higher the intensity of the rumble, or the faster the counter weight spins. In order to

capture this intensity value, the variable voltage can be measured and used to identify the current state of each of the vibration motors.

In this case, a jumper wire was attached to the positive terminal of each motor and extended through the controller case. During installation of the jumper wires it was noted that the vibration motor on the right of the controller spins counterclockwise, and as a result the \pm terminals were swapped in the factory controller. To obtain the positive voltage values, the jumper wire was attached to the negative terminal.

An interface board is required to translate the varying voltage values into something that can be processed by a computer for transmission to a remote device. In this case, an Arduino Duemilanove board was used (Fig. 1). This board contains 6 analog inputs and an USB interface to communicate data to a computer. This provides a standard interface and more than enough analog inputs to handle the 2 values needed to represent the state of the vibration motors contained within the Xbox 360 controller. This board has enough inputs available to capture haptic data from three Xbox 360 controllers. Although this paper presents results using an Xbox 360 controller, the standard Xbox One, Playstation 3 and Playstation 4 controllers also contain 2 vibration motors and a similar modification can presumably be performed to capture data from those controllers as well.

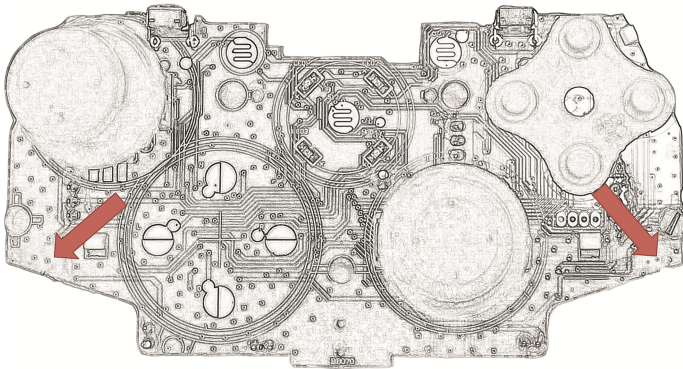


Fig. 1. Indirect haptic capture - read haptic data from XBox 360 controller. Arrows indicate solder points to retrieve left and right haptic values.

The arduino board contains custom firmware which is responsible for reading the values of the analog inputs and pass those values on to the host computer via USB for further processing. The firmware is a simple small infinite loop that reads the analog inputs every 10 ms, formats them in a comma separated string, and then writes them out the USB port. Through basic testing, it was discovered that each of the vibration pins sending voltage to the motors behaved in a similar way when the motor was vibrating vs idle. The arduino board contains a 10 bit analog to digital converter. When the vibration motor was idle, the analog pin on the arduino board reported back a value of 1024. When the motor was at its fastest, the analog pin contained a value of 10. This was the same for both motors within the Xbox 360 controller. Thus a simple formula was created to take those values and normalize them to the range of 0 (no activity) and 1 (full speed).

This was done to eliminate any variation of non-standard values that different interface boards may report back to the host.

6 Haptic Helper Service

The Haptic Helper Service (HHS) is responsible for taking the data from either the direct or indirect haptic capture methods and transmitting them to a remote server such that remote devices can receive this information and replicate the haptic state of the device. Messages passed to the HHS describe the behavior of the haptic motors. The message includes an index of the controller, an index of the haptic motor, and a value of the intensity. This message is passed to the HHS on an as needed basis. Whenever the haptic value changes for a particular haptic motor, a message is passed to the HHS.

In order to prevent a run away haptic situation, the game program is required to ping the HHS every 1000 ms in order for the HHS to acknowledge that any lapse in communication from the game is intended.

All messages include a sequence number, which the game program is responsible for incrementing. The HHS will acknowledge every message by sending back an ACK or a NACK, along with the sequence number. If the game program receives an ACK or NACK with an incorrect sequence number, it must retry the message until an ACK is received. Once the HHS has ACK'd the message, the game program no longer needs to keep this message in its buffer.

The capture methods present data in the format of a comma separated string representing up to six different haptic motors. This allows up to three controllers worth of data in the format Controller1Left, Controller1Right, Controller2Left, Controller2Right, Controller3Left, Controller3Right. The values of each of these are floats within the range of 0.0 (not active) to 1.0 (fully active). The HHS is configured with the URL of the remote HTTP server and when started, it has the option to create a new session ID or use an



Fig. 2. Complete haptic capture controller modification with arduino board

existing session ID. The session ID allows for multiple haptic streams occurring simultaneously as the client must identify the session ID to obtain data from the desired game play session.

7 Methods

Client Real Time Streams. The transmission of the data in the functioning prototype is through a simple protocol using HTTP posts passing each variable as a separate parameter to the post. The variables include the session ID, controller ID, and the six motor state variables. When a client receives the response from the HTTP post, it must assume that the values returned are the real time values of the haptic motors. As a result, the client must poll often to retrieve updates in real time. When using haptic relay locally, the client expects to receive the data in real time such that the delay between when the player feels the vibration and when the spectator feels the vibration is minimal. As a result, the message to the local controllers is sent at the same time as the message sent to the HTTP remote host, which makes those write at roughly the same speed. Any delay would come from the polling cycle of the client, which was set to fire off along with video frame updates at the rate of 30 times per second.

Using a hardware capture method introduces more delay as the intensity of the motors is also periodically polled at 100 times a second. Depending on network conditions, this could result in 10 ms delay on the capture, and a 34 ms delay on the client, with any additional delay occurring due to network traffic. Although there is some delay that may surpass 100 ms, it is negligible when paired with an online streaming session over a streaming network such as Twitch.tv. Twitch is typically delayed by 30 s. Even if there are lengthy delays on the haptic capture server, it is expected that any of these delays will be several orders of magnitude lower than the video delay presented.

In order to properly match up the haptic stream with the video, the client replay service has an option to delay the presentation of the vibration cues to the player, allowing for proper synchronization. In the prototype this is a manual process. The client viewer must adjust the slider from 0 to 120 s until he finds the value that is correct. No matter what the delay is, the client is always polling the server to receive the real time updates and queues them up until the presentation time has arrived.

The hammering of the server by the clients is not scalable. When not using a haptic capture device, it is possible to limit the amount of traffic. In this case, if the client software knows how long the vibration cue will be presented to the player, it can deploy a message to the server indicating the entire duration. For example, the game may know it will present a 5 s 100 % intensity cue on vibration motor 1. When the client reads this message, it knows that there is no need to keep polling for the next five seconds and will hold off repeatedly requesting information. This becomes much harder in the case of a hardware capture because the future of the haptic stream is not known and a polled method must be used.

The current prototype relies on streaming the haptic information through different servers than the originating video. Even after going through a manual sync process, the two streams can become out of sync. In order to avoid this issue, protocol modifications

are suggested to follow similar streaming techniques for closed captioning. Two standards for synchronizing text to video are described below and then an addition to these is proposed to utilize haptic information streams in a similar fashion.

WebVTT: The Web Video Text Tracks Format [22] was last updated at the end of October, 2014 and describes captions or subtitles for video and audio content distributed across the internet through HTML. The text tracks can be passed alongside a video by using the <TRACK> tag. WebVTT streams are in the following format:

```
00:11.000 ->00:13.000
<v Roger Bingham> We are in New York City
```

This entry in the stream indicates that from seconds 11 until seconds 13, the string shown in the second line should be displayed. Haptic information can be encoded in this exact same format. Using the same time encoding, the second line can contain information about the state of the motors during that time. A proposed addition to this specification would allow the definition of haptic states. The following encoding would indicate that the haptic motor on controller 0 with identifier 0 should vibrate at half of its capacity between seconds 11 and 13:

```
00:11.000 ->00:13.000,0,0.5
```

This could be extended to identify a sequence of haptic signals that could be performed over the time defined. For example, the following json encoding would indicate that vibration motors 0 and 1 should pulse at full intensity for 1 s, followed by a half a second of silence for the time period of 11 s through 21 s:

```
00:11.000 ->00:21.000 "pulses": [{"duration": 1, "pulse": [{"id": 0, "value": 1, "id": 1, "value": 1}, {"duration": 0.5, "pulse": [{"id": 0, "value": 0, "id": 1, "value": 0}]}
```

These types of enhancements to WebVTT will work best when the gameplay session is archived as all instructions will include a start time and end time. This type of encoding could potentially be used in the scenario where the client is streaming these commands in real time (as shown in the prototype) and the actual video delay is larger than the longest pulse. For example, if the video delay was 30 s, and the haptic stream was occurring in real time and depended on the client to buffer and present the vibrations to the player in synchronization with the video broadcast, as long as any piece of data did not take place longer than 30 s, the full haptic capabilities would be relayed to the player without any perceivable difference.

There are times when this might not be sufficient and directly embedding the haptic data into the video stream may be preferred. Text can be directly embedded into MPEG 4 video streams using the definitions shown in Part 17 and Part 30 of the standard. Part 30 defines the embedding of text as WebVTT into MPEG4 video streams. Enhancing Part 30 to include definitions for haptic streams will provide a convenient mechanism for packaging video and haptic together, however it still may suffer from some of the issues that come with requiring to know the start time and end time of a piece of haptic information. Part 17 defines the use of encoded text as 3GPP Timed Text [23]. This text

standard embeds the text into the video stream at the appropriate times. This can be enhanced to embed haptic status whenever the status changes. It will create a low overhead as nothing will be polled, and the stream will not be filled with repetitive information. It will allow the live broadcasting to continue without the need to synchronize time with the live stream and the haptic stream as they both will be carried in the same structure.

8 Limitations

Client Haptic Reproduction. There may be hardware and software limitations when the client attempts to reproduce a haptic stream. If the broadcaster and the receiver are using the same hardware, then the stream should be replicated in very near real time. An issue with replicating the stream becomes important when the client is either using a different hardware controller than the broadcaster, or the client is using a different hardware controller than the original software was designed for.

Different types of controllers have various types and number of motors. The Xbox 360, Xbox One, Dual Shock 3 and Dual Shock 4 are all similar in that they each have two vibration motors in each controller. That should make haptic broadcasts designed for these controllers and replayed by these controllers interchangeable for the most part. They do use different types of motors so a player broadcasting a haptic capture of an Xbox 360 controller may feel slightly different to a player who is receiving the stream with a Dual Shock 4 controller. Hardware limitations may be present when the hardware capabilities of the controllers differ significantly. For example, if a hardware capture of an Xbox 360 controller is being replayed by a player who is holding a Wii Remote, the client side may lose some of the detail that the player is feeling. The Wii Remote only contains one vibration motor and it may not be able to replicate the same feeling as two vibration motors. The same issue goes for someone who is using a cell phone as the client. Phones typically also only have one vibration motor and may not be able to replicate the stream accurately.

There are also software issues when attempting to replicate a haptic stream. Certain devices may be locked out of certain haptic features. The most common feature that clients may lack is in the replication of the intensity of the vibration. iPhones and Wii Remotes generally only allow either an ON or OFF setting with no varying intensity being available to developers. This differs greatly from the Xbox and Playstation controllers as they can vary the intensity and duration as needed. The current iPhone SDK even limits the duration of the pulse of available to developers to 1 s. Techniques such as starting and stopping the motors prior to them getting to their full speed may produce a feeling of different intensities; however, these techniques are unavailable in the default SDKs.

Capture Device Limitations. The current hardware prototype for capturing haptic signals from controllers has a limitation. The prototype shares a common ground between the controller and the arduino board. This allows for a simple and quick circuit to be developed that involves only the positive terminal of the vibration motor being connected to the analog inputs on the arduino board. If the controller and the capture

device were connected to different ground sources, the circuit may have to be slightly more complex. For example, if the shown circuit was used to capture the haptic signals from an Xbox One controller powered by batteries, the circuit would have to be modified. All tests performed with the haptic capture board for the purposes of this paper were performed with the arduino board and the controller both powered by the same USB source.

9 Future Work and Conclusion

The work presented here demonstrates the need and feasibility of transporting haptic information along with video streams of games. Future work will involve implementing the suggested haptic channel encodings into video transmission standards and to evaluate their performance. Performing a round trip test using a standard video streaming site, such as Twitch.tv, may require assistance from Twitch as they may re-encode videos on their end to maintain end user expectations, and during this re-encoding process the new haptic information may be lost.

In addition to games, this haptic encoding may be able to take movie viewing to the next level. Smart phones with haptic capabilities are common place, and it may be useful to consider using that personal technology while watching a movie. For example, if movie viewers were able to log into a haptic stream for whatever movie they were watching, and when an explosion on the screen occurs, a phone in the user's own pocket would vibrate giving the viewer a directed cue about onscreen activities. This paper presents a prototype and suggested enhancements to video transmission standards such that haptic information is also included in video streams. It can be used for enhancements for online video game streams.

References

1. Johnston, C.: Rumble Pak Titles On the Rise, GameSpot.com 23 May 1997. <http://www.gamespot.com/articles/rumble-pak-titles-on-the-rise/1100-2466717/>. Accessed 10 Jan 2015
2. maru-chang, SCPH, maru-chang.com 8 February 2004. <http://maru-chang.com/hard/scph/index.php/english>. Accessed 10 Jan 2015
3. Tremblay, M.R., Yim, M.H.: Tactile feedback man-machine interface device US Patent US6088017 A. Filed 24 April 1998, 11 July 2000
4. Tremblay, M.R., Yim, M.H.: Tactile feedback man-machine interface device US Patent US6424333 B1. Filed 18 April 2001, 23 July 2002
5. Jenkins, D.: Immersion Wins Sony DualShock Patent Infringement Suit. GmaSutra.com, 22 September 2004. <http://www.gamasutra.com/view/news/95287/ImmersionWinsSonyDualShockPatentInfringementSuit.php>. Accessed 15 Jan 2015
6. Riddell Williams: Microsoft Corp. Vs Immersion Corp., 18 June 2007. <http://blog.seattlepi.com/microsoft/files/library/immersionmicrosoft.pdf>. Accessed 9 Jan 2015
7. Skillman, A.: Immersion Issues Patent Licenses for Tactile Feedback Video Game Console Peripherals, 11 July 2007. <http://ir.immersion.com/releasedetail.cfm?ReleaseID=253387>. Accessed 12 Jan 2015

8. Fitzgerald, D., Wakabayashi, D.: Apple quietly builds new networks. *Wall Street J.* (2014)
9. Eadicicco, L.: 10 Facts About Twitch, The Company That Amazon Is Buying, That Will Blow Your Mind. *Business Insider*, 25 August 2014. <http://www.businessinsider.com/statistics-about-twitch-2014-8>. Accessed 12 Jan 2015
10. Amazon.com. Amazon.com to Acquire Twitch. Amazon.com Press Release, 25 August 2014
11. Grundberg, S., Hansegard, J.: YouTube's biggest draw plays games, earns \$4 million a year. *Wall Street J.* (2014)
12. PewDiePie, PewDiePie - Youtube. <https://www.youtube.com/user/PewDiePie>. Accessed 20 Jan 2015
13. Charbonneau, E., Hughes, C.E., LaViola Jr., J.J.: Vibraudio pose: an investigation of non-visual feedback roles for body controlled video games. In: *Sandbox 2010*, Los Angeles, California, 28–29 July 2010
14. Hamam, A., El Saddik, A., Alja'am, J.: A quality of experience model for haptic virtual environments. *J. ACM Trans. Multimedia Comput. Commun. Appl. (TOMM)* **10**(3), Article No. 28 (2014)
15. Nesbitt, K.V.: Multi-sensory game interface improves player satisfaction but not performance. In: *AUIC 2008 Proceedings of the Ninth Conference on Australasian User Interface*, vol. 76, pp. 13–18 (2008)
16. Alahakone, A.U., Senanayake, S.M.N.A: Vibrotactile feedback systems for rehabilitation, sports and information display: a review. In: *International Conference on Advanced Intelligent Mechatronics, AIM 2009*, pp. 1148–1153 (2009)
17. Xu, Z., Yu, H., Yan, S.: Motor rehabilitation training after stroke using haptic handwriting and games. In: *iCREATE 2010, Proceedings of the 4th International Convention on Rehabilitation Engineering Assistive Technology*, Article No. 31 (2010)
18. Morelli, T., Foley, J., Column, L., Lieberman, L., Folmer, E.: VI-Tennis: a vibrotactile/audio exergame for players who are visually impaired. In: *FDG 2010, Proceedings of the Fifth International Conference on the Foundations of Digital Games*, pp. 147–154 (2010)
19. Morelli, T., Foley, J., Folmer, E.: Vi-bowling: a tactile spatial exergame for individuals with visual impairments. In: *ASSETS 2010, Proceedings of the 12th International ACM SIGACCESS Conference on Computers and Accessibility*, pp. 179–186 (2010)
20. Morelli, T., Foley, J., Lieberman, L., Folmer, E.: Pet-N-Punch: upper body tactile/audio exergame to engage children with visual impairments into physical activity. In: *GI 2011, Proceedings of Graphics Interface 2011*, pp. 223–230 (2011)
21. Kendzierski, D., DeCarlo, K.: Physical activity enjoyment scale: two validation studies. *J. Sport Exerc. Psychol.* **13**, 50–64 (1991)
22. WebVTT, WebVTT: The Web Video Text Tracks Format. WebMedia Text Tracks Community Group, 30 October 2014. <http://dev.w3.org/html5/webvtt>. Accessed 12 Dec 2014
23. 3GPP, 3GPPSpecification detail. Spec26.245. <http://www.3gpp.org/DynaReport/26245.htm>. Accessed 20 Jan 2015