Design Research Seminar Report

Documentation of Mainstream VR Interactions

Vivek Paul Joseph
156330004
Interaction Design
M. Des (2015-17)

Guide: **Prof. Jayesh Pillai**

IDC School of Design
Indian Institute of Technology Bombay

Approval Sheet

The Interaction Design DRS (Design Research Seminar) entitled “Documentation of Mainstream VR Interactions” by Vivek Paul Joseph, Roll Number 156330004 is approved, in partial fulfillment of the Master in Design Degree in Interaction Design at IDC School of Design, Indian Institute of Technology Bombay

Guide:

Declaration

I declare that this written document represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Vivek Paul Joseph
156330004

IDC School of Design,
Indian Institute of Technology Bombay

March 2017

Aknowledgements

I would like to thank Prof. Jayesh Pillai for his support and guidance through the duration of this project. I am grateful to Prof. Girish Dalvi, Prof. Ravi Poovaiah, Prof. Venkatesh Rajamanickam and Prof. Anirudha Joshi at IDC for their valuable
feedback and suggestions.

Table of Contents

Abstract

Today, there are a variety of devices/platforms that are capable of providing virtual reality (VR) experiences. Some examples are: google VR (Cardboard & Daydream), MS Hololens, Oculus Rift, htc Vive, etc. Each device brings in an opportunity for a unique way of interaction between the user and the system. This largely depends on the hardware capabilities and limitations of the the device itself and of course, the imagination of the designers/developers developing applications for these devices.

This study aims to explore the different kinds of interactions that are currently being employed by designers/developers across a variety of applications across different platforms.

Introduction

Motivation

Currently there are a lot of platforms that support VR interactions. Each platform has its own pros and cons; and specific interactions (with the player) that the platform can support. Hence, the platform to develop a VR experience for becomes an important decision for a designer. The designer can either select a platform and design an experience for the platform or the other way round (design the experience and select the platform, if it exists, that can support all of the interactions required to develop the application).
A repository of all the interactions (currently available) and the platforms that support them would be helpful to a designer to make the aforementioned choice, which is the motivation behind this exercise. There are other factors that are also to be taken into account while selecting the platform. Eg: Cost, hardware, ergonomics, availability, etc. Since such information is already available on the internet in the form of detailed reviews of the platforms, these factors outside the scope of this exercise, which will be targeted specifically towards the interactions that are supported by the platform.

Objectives

- Document the different types of interactions that are currently being utilized by apps for the major VR platforms
- Provide the collected information in the form of a digital reference guide for designers to decide which VR platform to develop for.

2. Data Collection

2.1 Approach

The approach followed here can be split up into the following steps:

1) Explore and identify the different major VR supporting platforms (Eg: Oculus Rift, htc Vive, etc.). It is to be noted that only the platforms that are out in the market have been considered, since there is little reliable information available on others.
2) Explore the currently utilized/supported VR interactions across all the identified platforms (based on apps/demos that have been released till date)
3) Arrange the collected information such that it can be used as a reference to help a designer decide what VR platform to work on

2.2 Major VR Platforms

The VR platforms considered under this exercise are:

- Daydream VR
- Gear VR
- Oculus Rift
- HTC Vive
- Playstation VR

Daydream VR
Developer: Google
Base Platform : Android
Hardware:

1. Headset
2. Wireless controller
3. Daydream supported android phone (display)

Other notable features:

- Untethered (Unrestricted head movement)
- Supports rotational tracking (from the phone) of head; and
  (approximate) positional and rotational tracking of the remote controller
- The phone acts as the display and processing unit for the apps
- Controller can be used as a pointer
- Controller has touchpad and buttons as further input modes

Gear VR
Developer: Samsung
Base Platform : Android
Hardware:

1. Headset
2. Gear VR supported android phone (Display)

Other Features:

- Untethered (Unrestricted head movement)
- 96°/101° Field of View (Device dependent)
- Compatible with (Galaxy Note 5, Galaxy S6/S6 Edge/S6 Edge+, Galaxy S7/S7 Edge)
- The phone acts as the display and processing unit for the apps
- The headset has a touchpad and back button on the side
- Supports rotational tracking and input from the touchpad and back button

Oculus Rift
Developer: Oculus VR
Base Platform: PC
Hardware:

1. Tethered Headset (with built-in screen)
2. XBox One Controller
3. 2 Oculus touch controllers (one for each hand)
4. 2/3 IR Sensors

Other notable features:

- Tethered to PC (Which acts as the processing unit)
- Individual screens for each eye (in the headset)
- Camera passthrough (using Leap Motion) - For the player to see
  Their surroundings without taking off the headset
- Player defines playspace during initial setup
- Positionally and rotationally tracked headset and touch controller

HTC Vive
Developer: HTC and Valve
Base Platform: PC
Hardware:

1. Tethered Headset (with built-in screen)
2. 2 Remote Controllers
3. 2 Base stations (sensors) - For player movement tracking

Other notable features:

- Individual screens for each eye
- Room scale Tracking
- 6 Degrees of freedom head tracking
- Player and hand positional tracking

PS VR

Developer: Sony IE
Base Platform: PlayStation 4
Hardware:

1. Tethered Headset (with built-in screen - Visor style)
2. PS Move controllers
3. PS4 Controllers

Other notable features:

- 100° Field of View
- 6 DOF Head tracking
- Player positional tracking
- Controller 6 DOF tracking

2.3 VR Interactions Used Across Platforms

The different interactions used across different platforms has been summarized into the following list:

1. Gaze : Look around
The view orients based on the orientation of the player's head. This is a basic VR interaction that is used in almost all VR applications. Can be either in 1st person or 3rd person.

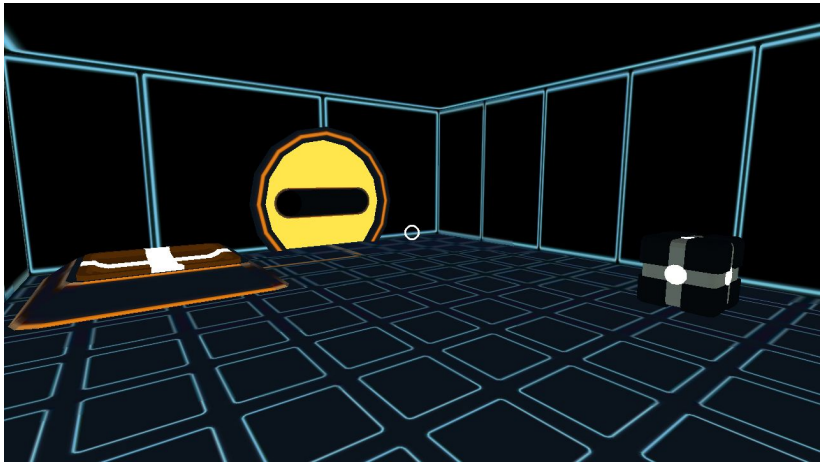
Supporting VR platforms: All

3rd Person: Lucky's Tale

1st Person: Dreadhalls

2. Gaze : As a trigger
A reticle provided at the center of the view used as a trigger for events upon targeting specific objects. The trigger is usually time based, i.e., the trigger activates only when the reticle remains on the object for a minimum threshold time period.

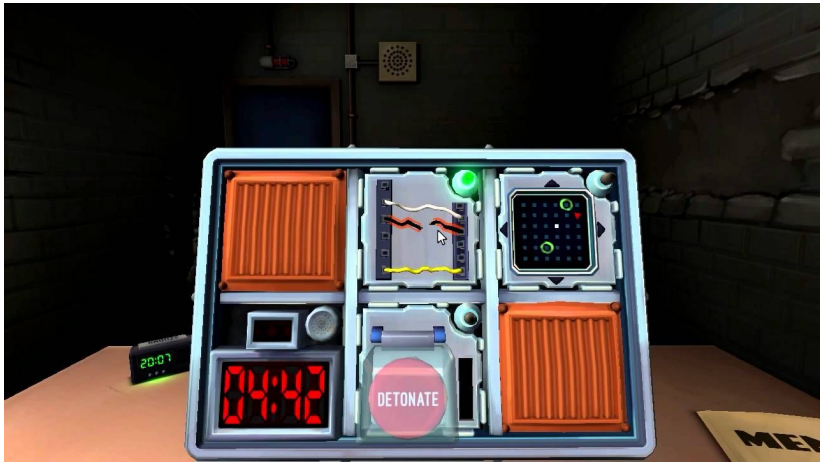
Supporting VR platforms: All



Eg: Gravity Pull (for cardboard VR)

3. Gaze, Click and Hold
Supported platforms: All platforms

Player orients gaze at an object. Clicks and holds an external button for a minimum time period to trigger action on the object. Here the player can look away from the object after the initial click and hold; and the action gets broken only if the player releases the hold before time.



Eg: Keep Talking and Nobody Dies (Gear VR)

4. Look down to reset view
Supported platforms: All platforms

Player looks down to reset the view (The direction that the player is currently looking at is set as front). This is especially helpful in apps where the player has to turn around a lot.

5.Automatically reorient objects to remain in front of player
Supported platforms: All platforms

Objects remain in front of the player's view regardless of where they are looking. If the turn their head, the object moves with them. This can be used for displaying text messages to the player.

Eg: Blu-VR (on Gear VR)

6. Controller: Point and Click

Supported platforms: Daydream VR, Oculus Rift, PS VR, HTC Vive

Applies to platforms which have a controller which can be used like a pointer to aim and trigger actions like zoom, shoot, select, etc.. This interaction is mostly used in two ways, that can be understood from the below examples:
i) Controller held like a wand

Eg: Cosmic Trip (Oculus Rift)

ii) Controller held like a gun

Eg: Arktika 1 (Oculus Rift)

7. Controller: Click, Hold and Move
Supported platforms: Daydream VR, Oculus Rift, PS VR, HTC Vive

Player points towards an object using the controller, clicks and holds the button to lock onto the object. Then, the player moves the controller around to manipulate the object.

8. Touchpad: Click, Hold and Rotate head
Supported platforms: Gear VR

Similar to #4. Player clicks the clicks the touch pad, holds it down and moves head around to manipulate the object.

9. Controller: Chop Action with Controller
Supported platforms: Daydream VR, Oculus Rift, PS VR, HTC Vive

Player does chop action with the controller to trigger some activity within the application. Generally used for global actions (which doesn't require any targeting). Eg: Reload weapon for shooting games

Eg: ArcSlinger VR (Daydream VR)

10. Controller: Tilt to Direct
Supported platforms: Daydream VR, Oculus Rift, PS VR, HTC Vive

Player tilts the controller to orient or direct an object.

Example: VR Carts : Daydream VR (Racing Games).

11. Controller: As a Gamepad
Supported platforms: Oculus Rift, PS VR, HTC Vive

Standard controllers (XBox and PS4 controllers) used as an input method. Similar to regular video games.

PSVR is a special case here. The controller here is positionally tracked and represented in the virtual space to scale. This helps player keep track of button locations.

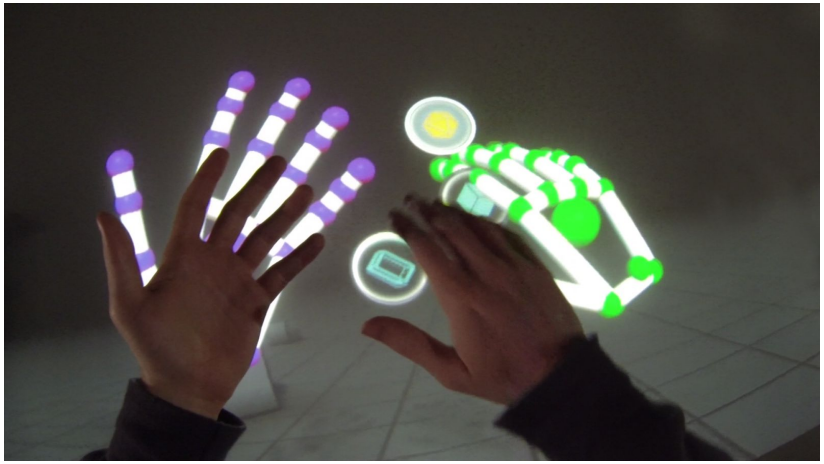
Img Source: highdefdigest.com

Eg: Playroom VR (PS VR)

12. Controller: Hand Gestures
Supported platforms: Oculus Rift

Basic hand gestures like thumbs-up and pointing with finger can be detected by Oculus Rift Touch controller. This introduces a basic hand presence to the experience. These gestures can be used to trigger certain actions within the VR application.

13. Gestures Control
Supported platforms: Oculus Rift & HTC Vive
(Requires Leap Motion add-on)
With the Leap Motion controller mounted on the head mounted device, the user's hand and finger movements can be tracked. There have been a few demo apps that demonstrate the capabilities of the device. Fully functional apps that utilize this are yet to be released.



Img Source: The Verge

14. Body Motion Tracking

Supported platforms: Oculus Rift, HTC Vive & PS VR

User can move around a defined playspace and be body tracked within that space. This is generally done using multiple IR cameras.

Img Source: blog.turbosquid.com

15. Touchpad: Swipe
Supported platforms: Gear VR
The touchpad on side of the headset on Gear VR allows for interactions triggered by swipe motion. Swipes in different directions can be mapped to different actions.

3. VR Tool Picker

VR Tool Picker is a tool for designers to select which VR platform to develop for based on the interactions that they want to use in the application. The designer is presented a list of interactions from which they select the ones that they want to use. The tool displays the platform that can support the selected interactions and also the list of hardware that would be needed to be used to enable these interactions.

There would be two modes to this tool:

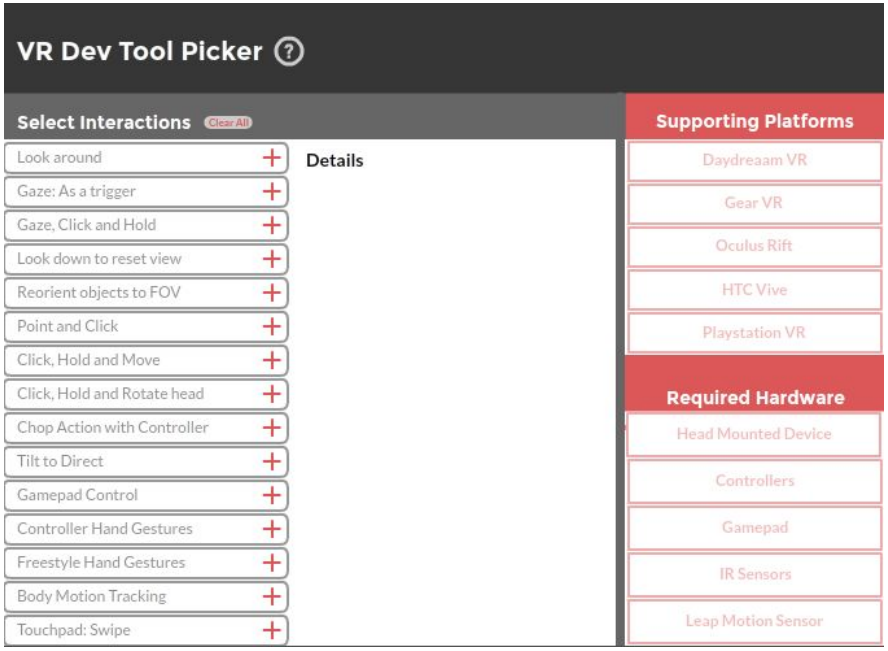
1. The user inputs the interactions that they want to incorporate in their application and the VR tool picker returns the different platforms that can support them and the hardware that would be needed for the same
2. The user inputs the platform that they want to work with and the VR tool picker returns the different interactions that the platforms can afford

For the scope of this exercise, Mode 1 will be designed and prototyped. Design-wise Mode 2 would also be more or less the same design elements, but with some layout changes.

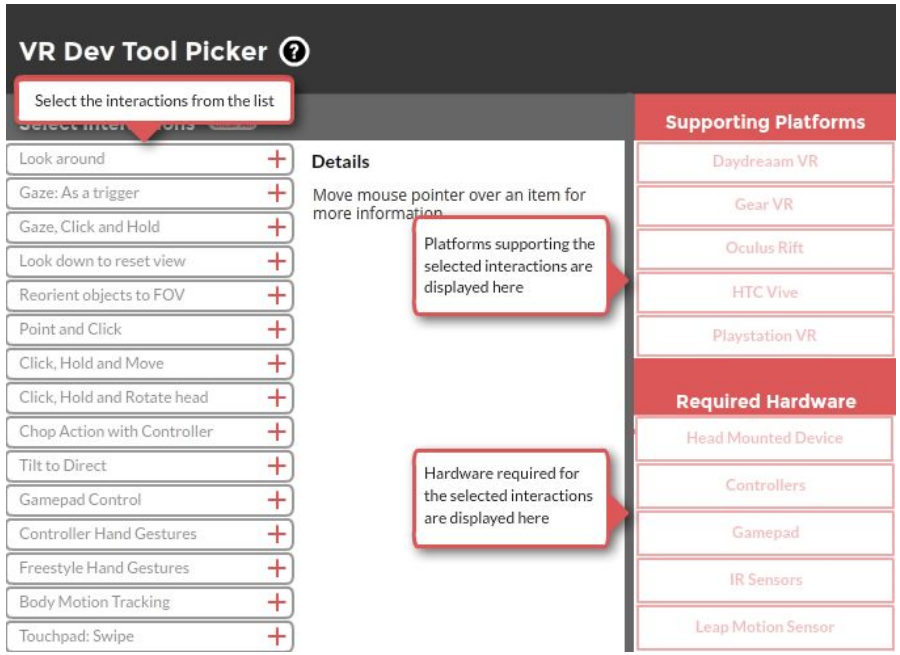
The attempt here was to convey the information clearly, have an easy-to-follow mental model (so that the user can easily pick up and use the tool) and to keep scalability in mind. Scalability is important for the tool to be useful in the long run, since new interactions, tools and platforms are being developed; and they all need to be incorporated in the tool to keep it relevant.

Different layouts and flows were tried out and based on the above mentioned guidelines, the one that followed it best was selected.

The data collected in the previous stages were collated into one database. Then different layouts and and each interaction was saved along with their supporting information (platforms; required hardware; and examples). The database was then used to generate the below layout. This approach helps maintain the scalability of the entire system.



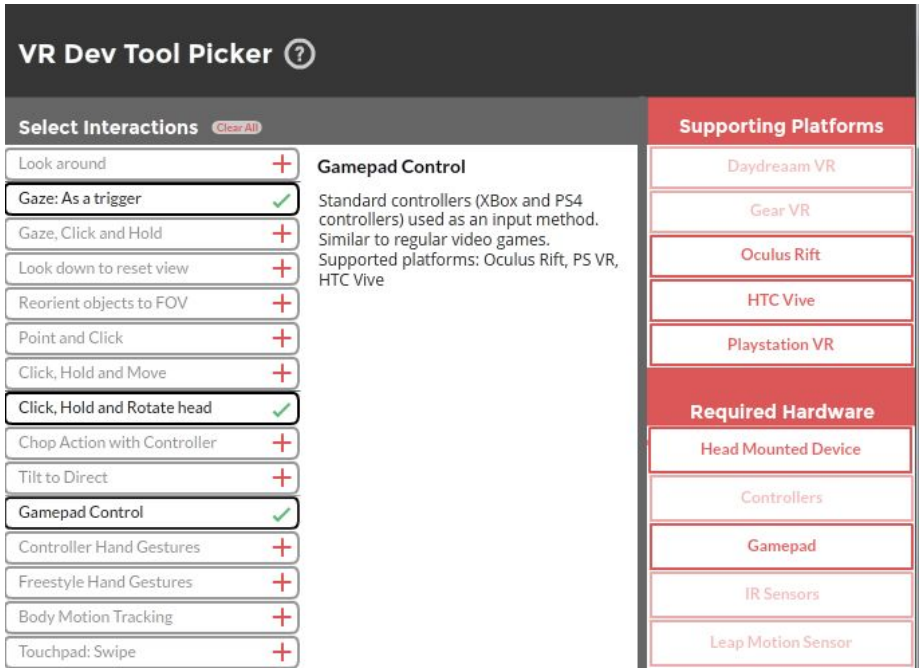
Layout Screen



Help Screen : To explain the working to user

Working of the system:

1. User selects the the interactions from the list on the left. (Multiple interactions can be selected)
2. The middle column displays data regarding the different interactions that the user hovers over
3. The lists on the right (in red) display the platforms that support the interactions and the hardware required to run it respectively.



Future Scope

Currently, only one of the two planned modes has been implemented under the scope of this project. The other mode would be equally useful and important to the user.

In order to achieve sustainable scalability, a few features need to be introduced :

- The database should be exported and referenced to an external file for easy access for editing/updating of the database as and when needed
- Currently the items on the list were created manually and the information applied onto them procedurally. Procedural generation of items should be introduced to support the previous point.

A visual representation of the interactions would help the user understand and visualize the interactions better. This can be done either using animations or live action videos as references.

Conclusion

Under this project, different interactions afforded by the major VR platforms (as on date) were documented and packaged into a reference tool for beginner VR application designers. While, in the current form, the tool can be used for the time being, it would need some modifications and updates in order to stay relevant in the future (to keep up with the new devices, platforms and interactions that get introduced).

References

"Cite A Website - Cite This For Me". *Wired.com*. N.p., 2017. Web. 18 May 2017.

"Daydream Believer: The 15 Best Google VR Apps You Can Install Right Now". *CNET*. N.p., 2017. Web. 18 May 2017.

"Medium – Read, Write And Share Stories That Matter". *Medium*. N.p., 2017. Web. 18 May 2017.

"Tested". *YouTube*. N.p., 2017. Web. 18 May 2017.

Image Resources:

http://www.droid-life.com/wp-content/uploads/2016/10/Daydream-VR.jpg
http://www.samsung.com/uk/consumer/mobile-devices/wearables/gear/SM-R323NBKABTU/images/gear-vr_supreme_vr.jpg
http://core0.staticworld.net/images/article/2015/09/oculus-touch-2-100616982-orig.png
https://www.destructoid.com/ul/349069-/psvr%201080p-noscale.jpg
https://31tnhx1wvgbv3b7i0lhz87x1-wpengine.netdna-ssl.com/wp-content/uploads/2015/12/img-features-mobile.png
https://www.petri.com/wp-content/uploads/2016/01/h169-1.jpg
https://www.destructoid.com/ul/350581-review-lucky-s-tale/AAA2-noscale.jpg
http://arvrdreams.com/wp-content/uploads/2016/03/Keep-Talking-02.jpg
http://www.samtregillus.com/wp-content/uploads/2016/08/gravitypull1.png
https://i.ytimg.com/vi/hcflnujcSYI/maxresdefault.jpg
http://funktroniclabs.com/presskit/cosmictrip/images/cosmictrip-screen1.png