

Desktop virtual reality in psychological research: a case study using the Source 3D game engine.

Steven L. Trawley
University of Edinburgh

Anna S. Law
Liverpool John Moores University

Matthew R. Logie
University of Edinburgh

Robert H. Logie
University of Edinburgh

While the use of virtual environments in psychological research is of growing importance, the high cost and lengthy development time is a likely deterrent to new users. In this article we argue that commercially available games platforms present an economical and accessible approach to creating high fidelity desktop virtual environments that are suitable for research purposes. We present a case study using the Source 3D game engine, a platform that is proving increasingly popular with researchers. We also describe a Python-based data extraction program that we have developed, which enables future users to rapidly and effectively process the large amount of data provided by this methodology. The evaluation of this program using data from a task created with this method (Logie, Trawley & Law, 2011; Trawley, Law & Logie, 2011) provides support for this approach. This freely-available Data Extraction Utility may remove one barrier to those wishing to join a burgeoning community of researchers utilizing desktop virtual reality to investigate important questions about human behavior.

Keywords: Virtual Reality, Methodology, Source Engine, Cognition

Introduction

Off the shelf commercial computer games have been used in several different areas of psychological research, from development of spatial cognition (Feng, Spence, & Pratt, 2007) to treatment of anxiety (Walshe, Lewis, Kim, O'Sullivan, & Wiederhold, 2003) and understanding the process of social influence in groups (Krotoski, 2009). This variety is perhaps not surprising given the effort made by designers and developers to create games that can motivate players to invest substantial time and effort into their completion. Of specific relevance to psychologists is the game genre known as the First-Person Shooter or FPS. These games represent a virtual environment as seen from the perspective of the player, whose avatar can be controlled via the mouse and keyboard or with a simple game controller. This is clearly an attractive proposition to researchers looking to expand their current methodologies into virtual reality (VR). Although the majority of computer games are based around proprietary closed source software with no modification potential, mak-

ing them unsuitable for the majority of research questions, several commercially available FPS games now come with their own software development kit (SDK) that offers researchers the ability to adapt the game for their particular research needs, on the condition that the use of the SDK is not-for-profit. The work by Smith and Trenholme (2009) is an excellent example of this approach. Using the Source game engine SDK supplied with the game 'Half-Life 2' they created a virtual replication of a university campus building to assess fire evacuation behavior among staff members. Participant behavior across three different fire scenarios followed a similar trend in both the real and virtual buildings. However, their sole method of data collection was manual scoring of video recorded during the test session (evacuation time for each scenario was the only variable reported). This is a typical solution to the problem of data collection when using commercial computer games in research (for a similar approach see Law, Logie, & Pearson, 2006). Although other approaches are available which can provide more detailed data output, such as the Quake game engine (Frey, Hartig, Ketzler, Zinkernagel, & Moosbrugger, 2007; Hartley, Trinkl, & Burgess, 2004), these require the user to understand the specific scripting language used for these game engines. The advantage of the Source engine is its ability to produce a detailed log file of the participants' movements and behavior without coding input from the researcher. Low-cost non-game alternatives that are designed specifically for research

We acknowledge Leverhulme Trust research grant number F/00 158/W which supported the research that we report here. Correspondence concerning this article should be sent to Steven Trawley stevetrawley@gmail.com or Robert Logie rlogie@staffmail.ed.ac.uk.

and require no advanced technical skills (such as the Maze Suite (Ayaz, Allen, Platek, & Onaral, 2008)), lack the advanced graphical capabilities such as the smoke or fire effects used by Smith and Trenholme (2009). They also lack the online community support found with a commercially successful computer game. More advanced low-cost 3D software options, such as Panda3D or Unity, require extensive programming knowledge. Similarly, the commonly used Vizard is expensive (\$3,990 USD) and requires knowledge of the Python programming language.

A full review of the application of game engines in psychology research is beyond the scope or aim of this paper (for benefits and comparison of game engines see reviews by Washburn, 2003; Glotzbach-Schoon et al., 2013; Gatzidis, Brujic-Okretic, & Baker, 2008; respectively). However, a literature search of the use of the Source gaming engine and its use in psychological research was undertaken using the Google Scholar and PubMed databases. The aim was to allow us to better understand how this engine is currently used, and more specifically, its replicability for interested researchers. The search was conducted in February 2013 and search terms included combinations of the following: psychology, Source engine, Half Life 2, virtual environment, and virtual reality. All retrieved articles were screened to omit any research not published in English and any design that did not concern participants interacting with a virtual environment (e.g., use of the Source engine to create virtual facial stimuli, Dyck, Winbeck, Leiberg, Chen, & Mathiak, 2010; or video footage of virtual museum, Miller & Carlson, 2011). This review of empirical work indicates that the majority of studies fail to describe or reference a description of how the Source engine data were collected (see Table 1).

It is also noteworthy that although small in number, the majority of these studies have only been published in the last two years, indicating an increasing interest in this methodology. Furthermore, the authors of these studies cover a wide range of psychology research areas, including human computer interaction, emotion, spatial cognition, multitasking, prospective memory, moral decision-making, and education. This demonstrates the potential of this approach and how it can enable researchers from a diverse background to use an otherwise inaccessible methodology. However, barriers still exist, for example, the absence of a computer science background in many members of the psychology research community. To the authors' knowledge, there exists no article or report that details the steps necessary to automatically collect positional and behavioral data from the Source engine. The work by Nacke et al. (2010) and Greville et al. (2013) represent the only examples that provide a complete description of their data logging methodology by virtue of a reference to an earlier technical paper (Nacke, Lindley, & Stellmach, 2008). However, this approach requires knowledge of C++ code to be successfully implemented. Other groups, such as Sturz

et al., (e.g. Sturz & Bodily, 2011 see Table 1 for additional references) have used a second computer running a Half-Life server from which they collected participant movement and behavioral data. However, in addition to the absence of documentation regarding this approach, the resulting data set is a complex log file with a large amount of redundant information. The combined problems of creating, summarizing and analyzing an automatically generated log file from the Source game engine are now addressed in this paper.

Objectives of the present study

In this paper we detail a straightforward method of using the Source game engine that automatically provides high fidelity data output to the researcher. This aspect distinguishes this approach from previous uses of this game engine (see Table 1). We have provided a data extraction utility (DEU) to facilitate data analysis, a procedure that requires no knowledge of scripting languages or complex adaptation of the Source engine C++ code to automatically collect behavioral data (e.g., Nacke et al., 2008). The DEU we have created presents the user with a graphical user interface (see Figure 5), which provides a simple and efficient method of extracting movement data from the large log file produced by the source game engine. In addition, it allows the user to identify specific areas of interest from which they want to extract movement data. The programming language Python was used to create the DEU as it can be downloaded for free, leaving the purchase of the game (approximately \$20 USD) as the only financial cost in the implementation of a desktop VR experiment. This DEU was created solely for the purpose of enabling researchers to use desktop VR, especially those researchers who have not used this approach before. To demonstrate the validity of this methodology and the analysis it affords we have selected a small sub-sample from a multitasking dataset previously reported elsewhere that used the Edinburgh Virtual Errands Task or EVET (Logie et al. (2011); Trawley et al. (2011)). The EVET was based on the Virtual Errands Task (VET) reported by McGeorge et al. (2001). One of the main motivations for the development of the EVET was the limited functionality of the original VET, both in terms of its graphical output and its reliance on post-test video analysis. In the next section, we describe how we used the Source game engine SDK to create this task and demonstrate the analysis it affords with the example dataset.

Method

Overview of the Source engine

The Source engine SDK tool Hammer is responsible for creating and editing three-dimensional game maps and is available free with the purchase of a Source engine game such as Half-Life 2. It is not our intention to detail every

Table 1
Psychological studies using the Source game engine.

Authors and Year	Behavior Measures from Source engine	Data collection Method	If data logged automatically was process documented?
Cenkner et al. (2011)	Avatar location / behavior 1Hz	Automatic log	No
Becker-Asano et al. (2011)	Avatar location	Unclear	No
Bodily et al. (2012)	Avatar location / behavior	Automatic log	No
Bodily et al. (2011)	Avatar location / behavior	Automatic log	No
Glotzbach-Schoon et al. (2013)	None	not applicable	not applicable
Greville et al. (2013)	Avatar location / behavior	Automatic log	Yes
Hartmann and Vorderer (2010)	None	not applicable	not applicable
Jeong et al. (2012)	None	not applicable	not applicable
Jouriles et al. (2009)	None	not applicable	not applicable
Kivikangas et al. (2011)	Avatar location / behavior	Automatic log	Yes
Kozlov and Johansen (2010)	Avatar behavior	Unclear	No
Law et al. (2012)	Avatar location / behavior 1Hz	Automatic log	No
Legge, Madan, et al. (2012)	None	not applicable	not applicable
Legge, Spetch, et al. (2012)	Avatar location / behavior	Unclear	No
Logie et al. (2011)	Avatar location / behavior 1Hz	Automatic log	No
McQuiggan et al. (2008)	Avatar location / behavior	Automatic log	No
Nacke et al. (2010)	Avatar location / behavior	Automatic log	Yes
Radvansky and Copeland (2006)	Avatar behavior	Unclear	No
Radvansky et al. (2010)	Avatar behavior	Unclear	No
Smith and Du'Mont (2009)	Avatar location / behavior	Automatic log	No
Smith and Trenholme (2009)	Avatar location	Unclear	No
Sturz and Bodily (2010)	Avatar location / behavior	Automatic log	No
Sturz and Bodily (2011)	Avatar location / behavior	Automatic log	No
Sturz and Diemer (2010)	Avatar location / behavior	Automatic log	No
Sturz and Kelly (2009)	Avatar location / behavior	Automatic log	No
Sturz et al. (2009)	Avatar location / behavior	Automatic log	No
Sturz and Brown (2009)	Avatar location / behavior	Automatic log	No
Sturz, Paul Cooke, and Bodily (2011)	Avatar location / behavior	Automatic log	No
Sturz et al. (2012)	Avatar location / behavior	Automatic log	No
Sturz et al. (2013)	Avatar location / behavior	Automatic log	No
Sturz, Gurley, and Bodily (2011)	Avatar location / behavior	Automatic log	No
Sturz et al. (2010)	Avatar location / behavior	Automatic log	No
Sutton et al. (2010)	Avatar behavior	Unclear	No
Talbot et al. (2009)	Avatar behavior	Post-test video analysis	not applicable
Tlauka et al. (2011)	None	not applicable	not applicable
Trawley et al. (2011)	Avatar location / behavior 1Hz	Automatic log	No

step here as a variety of excellent text and video based Hammer tutorials are already available on the internet that will facilitate rapid learning for interested researchers; indeed the large and vibrant online community support base is one of the advantages of using this software. Moreover, the specific technical detail we wish to highlight in this paper is how automatic data collection from the Source engine can be

achieved without having to adapt the complex Source engine C++ code. As far as the authors are aware, this is the first documented report of how this can be achieved. Before we continue, however, a brief description of how Hammer works is necessary.

The Hammer 3D editor consists of 4 screens and several control panels (see Figure 1). These screens allow the user

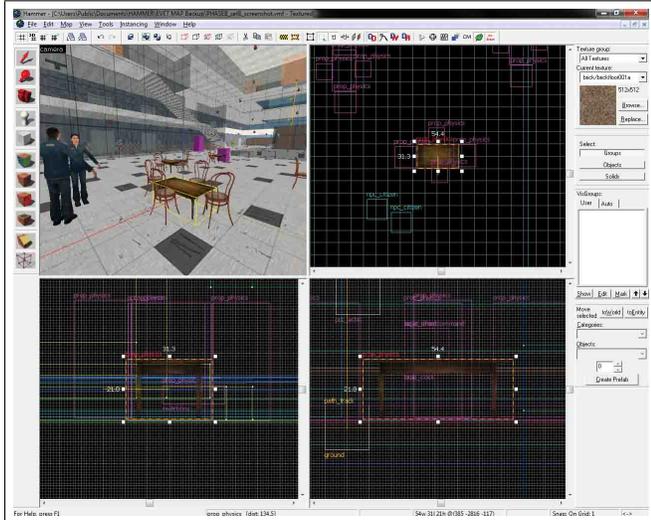


Figure 1. The basic four screen view of Hammer level editor.

to create and navigate through a virtual environment. In Figure 2 the icon panel on the left provides a choice of several functions, such as basic object creation (e.g., for creating walls and doors), applying textures, and editing tools to adapt how the objects appear (e.g., cutting doorways into walls). In addition, the Hammer editor provides specific functions for game control. Two of these are essential for automatic collection of behavioral data within the virtual environment. The first is called the point client entity, which can issue specific commands to the game engine. The command that is of interest here is the status command, which prints information about the current 3-D map to a text file. Each time the point client entity receives this command it will log the positional coordinates of the gamer position and the corresponding game time. The second function that we need is the logic timer entity, which forwards a message at an interval set by the user. By asking this function to fire regular status commands to the point client entity we can produce a time stamped recording of participant movement in the virtual environment during the test. Furthermore, specific activities that are of interest to the researcher, such as picking up an object or entering a specific location, can be linked to the point client command thereby recording that activity and the time it was performed. An example of this activity recording is shown in Figure 2, where the object property dialog box on the left shows two rows of information about a movement trigger (the transparent rectangular box in top right of picture). The first row records the moment when the participant approaches the noticeboard and enters the trigger box (this screenshot was taken from the EVET hammer file - see below).

The second row displays an on screen message that the participant has completed a specific errand. In the example here the errand required the participant to read a no-

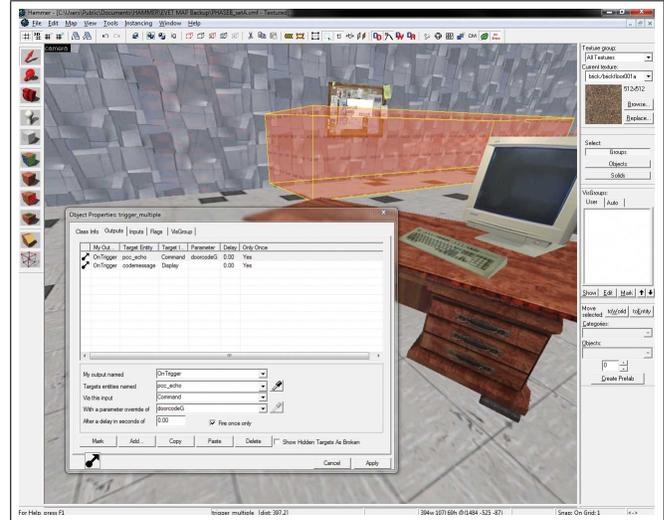


Figure 2. Hammer screen showing the trigger entity (orange box) and its properties box.

ticeboard sign. In order to automatically record this event the first line is crucial as it records this in the data file by linking the trigger to a point client entity named 'pcc_echo' and asking it to provide a command with the parameter of 'doorcodeG'. The end result of this can be seen in the datafile shown in Figure 3. Each block of data contains both the time and positional coordinates of the participant. The highlighted block shows a recording of 'doorcodeG', detailing the time when this errand was performed (in this example 00:15 seconds into the test). This file is known as the console.log text file and is located in C:\Program Files\Steam\steamapps*username*\half-life2\hl2 (a shortcut can be made to the desktop for ease of use). To enable the creation of this file the command '- condebug' should be added as a launch option in the steam menu properties box of the game being run. At the end of each session this text file can be saved under a different name and then the original data can be deleted in preparation for the next test point. However, new data are always appended to this file, so no data are lost if the researcher forgets this step at the end of an experimental session.

Validation of the methodology

To illustrate the utility and validity of our methodology, we next describe how our approach can accurately record participant behavior in a virtual environment and highlight specific behavioral trends across groups. To achieve this we have extracted the top and bottom quartile of performers from a study conducted by the authors that used this technique to create a test of multitasking, the EVET (Logie, et al., 2011; Trawley et al., 2011). Using this subset of high and low performers for demonstration purposes we will de-

```

921 version : 1.0.0.0/15 4295 insecure
922 map      : PHASEB_setA at: 1298 x, -454 y, -63 z
923 players : 1 (1 max)
924
925 # userid name uniqueid connected ping loss state adr
926 # 2 "Mem" STEAM_0:0:0 00:15 15 0 active loopback
927 Unknown command: doorcodeG
928 hostname: Half-Life 2
929 version : 1.0.0.0/15 4295 insecure
930 map      : PHASEB_setA at: 1313 x, -455 y, -63 z
931 players : 1 (1 max)
932
933 # userid name uniqueid connected ping loss state adr
934 # 2 "Mem" STEAM_0:0:0 00:15 15 0 active loopback
935 hostname: Half-Life 2

```

Figure 3. Example console.log data file containing both positional and event data (highlighted in bold).



Figure 4. Screen shot of the EVET showing the main concourse and the two stairwells.

tail how such a large amount of data (an eight minute EVET test will generate approximately 4800 three-dimensional positional data points) can be effectively analysed and summarised. We will begin with a short overview of the EVET followed by a description of our data extraction program and the analyses afforded by the dataset it creates.

The Edinburgh Virtual Errands Task (EVET)

The EVET virtual building was assembled and modified using the Hammer editor builder (engine version Half-Life 2). It was rectangular in shape, with thirty eight rooms spread over four storeys. All floors could be accessed via two internal stairs located either side of a large central concourse. Rooms were numbered along with their respective floor, for example, ground floor rooms were labeled G1, G2 etc., first floor rooms were labeled F1, F2, and so on. Figure 4 shows a

screen shot of the concourse taken from the top floor. Participants moved around this virtual building using the keyboard and mouse. Five keys ('a', 'd', 's', 'w' and 'e') were responsible for player movement and physical actions. Visual pitch (up and down) and yaw (spin left and right) were controlled by the mouse. Using the technique described above, participant position and movement within the virtual building was automatically recorded as a series of XYZ spatial coordinates. By setting the logic timer entity interval rate to '.1' we achieved a sampling rate of approximately 10Hz. In addition, any actions made by the participants were recorded with a time stamp and automatically inserted into the data file (see Figure 3). Logie et al. (2011) assessed EVET performance based solely on this data collection method. All participants memorized an eight item errand list to criterion (see Table 2; set A or B was counterbalanced across participants). During the test phase participants had eight minutes to complete the errands in an order of their choosing. An EVET efficiency score was created for each participant that awarded points for errand completion and deleted points for rule breaks (for more information see Logie, et al, 2011). The top (N=41) and bottom quartile (N=37) performers were selected from the distribution of this score. The EVET was viewed on a 42-cm color monitor, and run on a Dell XPS PC with Intel Core Quad 2.33 GHz processor and 1 GB ATI Graphics Card.

Table 2
EVET errand lists (A & B).

Errands	Set A	Set B
1	Pickup Brown Package in T4 and take to G6	Pickup Computer in G4 and take to T7
2	Pickup Newspaper in G3 and take to Desk in S4	Pickup Milk Carton in T3 and take to Desk in F4
3	Get Keycard in F9 and unlock G6 (via G5)	Get Keycard in S9 and unlock T7 (via T6)
4	Meet person S10 before 3:00 minutes	Meet person F10 before 3:00 minutes
5	Get stair-code from notice board in G8 and unlock stairwell	Get stair-code from notice board in T10 and unlock stairwell
6	Turn on Cinema S7 at 5:30 minutes	Turn on Cinema F7 at 5:30 minutes
7	Turn off Lift G Floor	Turn off Lift T Floor
8	Sort red and blue binders in room S2. Sort as many binders as you can	Sort red and blue binders in room F2. Sort as many binders as you can.

Source engine Data Extraction

We have provided future users of the Source engine with a DEU written in the freely available Python language. The software and instructions for use can be downloaded freely from the EVET website along with the original EVET 3-D map files (www.psy.ed.ac.uk/people/rlogie/MT/index.html). The main purpose of this software is to extract the relevant data from the large log file created by the Source game engine (see Figure 3). The DEU will create (or append if they already exist) three text files. The first is named 'EVENTResults.txt' and contains all non-status commands that signify an action of interest and the time it occurred. Each list will be preceded by the file name that it came from. The second results file, 'DISTANCEresults.txt', contains a row for each data file extracted with four separate columns; file name, vertical, horizontal in Source engine units and total time on floor. The source measurement unit converts to 1.9 'virtual' centimeters. This conversion was applied for the dataset reported in this article. The third, 'XYZresults.txt' collapses positional data for all data files into three columns; x, y and z. This information can be helpful in creating graphical representations of participant movement (see Logie et al., 2011; Trawley et al., 2011). A fourth text file, 'Distance-ToLocation.txt' is created if the user enters values into the X and Y location boxes (see Figure 5). This file provides distance measurements from the participant's position in the virtual environment to a location designated by the user. This further broadens the usefulness of the DEU to other research scenarios using this approach (e.g., wayfinding research).

Results

A total of 78 students at the University of Edinburgh (47 women, 31 men), mean age 19.55 years ($SD = 2.45$, range = 18D32) were used in the analysis described below. This sample consists of the top ($N=41$) and bottom quartile ($N=37$) performers selected from the Logie et al. (2011) dataset. Our rationale for selecting these divergent performance groups was to provide a suitably contrasting dataset with which to demonstrate our methodology.

Example Data and Analyses Afforded

We describe below the rationale for, and the procedure followed to generate a variety of outcome measures from the sample dataset that would be impractical to extract using post-test video analysis of participants performing the task (e.g., Law et al., 2006; McGeorge et al., 2001; Talbot, Legge, Vadim & Spetch, 2009). Even a relatively short testing session requires a time-intensive video coding period to provide a basic overview of participant movement. For example, Law et al. (2006) and McGeorge et al. (2001) used an earlier version of the EVET, which produced no automatic data file and

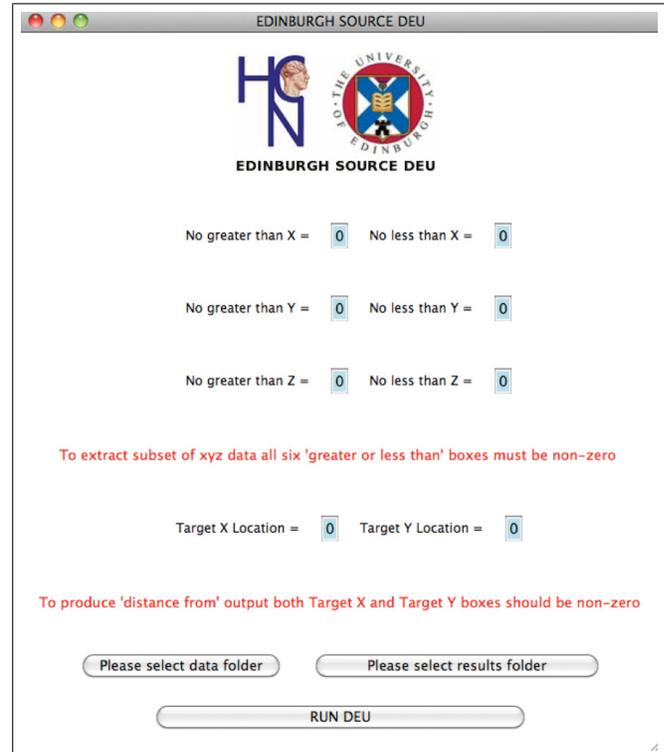


Figure 5. The Edinburgh Source Data Extraction Utility.

instead relied on a lengthy process of post-test video analysis. Even once completed, such post-test video analysis will only yield a broad qualitative assessment and a limited range of participant behavior measurements.

EVET distance travelled. This measure is part of the 'DISTANCE.txt' output described above. We have excluded any movements within rooms in order to directly index each participant's ability to efficiently navigate their path through the virtual building. The left and right partition walls in the EVET building were always positioned at 1168x and -401x grid coordinates at every floor. Excluding room positional data was achieved by removing any XY coordinates that exceeded these values via the data extraction software we have written (see Figure 5).

This measure shows no significant difference in meters travelled, $t(76) = 1.03$, $p > .05$, between lower ($M=653.20$, $SD=183.87$) and higher quartile EVET performers ($M=612.10$, $SD=166.20$). However, the distance measure can be made more sensitive to the building design by analysing movement on a floor-by-floor basis via adjusting the z-axis restriction box to focus on specific floors. This produces travel distance output on a floor-by-floor basis. These floors were categorised into two groups; single errand floors (set A: first and third; set B ground and second) and multiple errand floors (set A: ground and second; set B first and third). This adapted measure shows a significant

difference in meters travelled on single errand floors $t(76) = 4.71$, $p < .05$, with lower quartile EVET performers traveling further ($M = 310.88$, $SD = 183.87$) than higher quartile performers ($M = 226.63$, $SD = 37.13$). However, on the multiple errand floors there was no significant difference between lower ($M = 342.32$, $SD = 165.56$) and higher ($M = 385.47$, $SD = 147.97$) quartile performers ($p > .05$).

EVET travel time. In contrast to the above distance indices, this measure indicated the total amount of time each participant spent traveling in the EVET building. This measure is created by multiplying the number of positional data rows by the sampling rate (e.g., if the logic timer entity interval rate was set at '.1' multiple number of rows by .1). Similar to our adapted distance measure we excluded time spent in a room (i.e., completing a specific errand). The mean travel time for the two groups was significant even without taking into consideration the differences between multiple and single errand floors; a significant difference was found in travel time between lower ($M = 301.67$, $SD = 30.59$) and higher quartile EVET performers ($M = 254.03$, $SD = 35.92$). Using a similar approach to the distance measure we considered traveling time for single and multiple errand floors separately. As before, there was a significant difference on the single errand floor between lower ($M = 116.93$, $SD = 35.43$) and upper quartile ($M = 69.23$, $SD = 13.43$) performers ($t(76) = 7.92$, $p < .05$). There was no significant difference between lower ($M = 184.75$, $SD = 39.03$) and upper quartile ($M = 184.81$, $SD = 32.16$) performers on the multiple errand floors ($p > .05$).

Discussion

Our approach demonstrates an effective way of using off the shelf VR software to collect and summarize a large amount of behavioral data. This method allows a researcher to create a detailed record of participant movements and actions in a custom built virtual environment without any advanced technical skills or programming knowledge. A well documented automated data collection and extraction process is a crucial aspect of this methodology and avoids the lengthy process of video post analysis. It is this feature that distinguishes this approach from previous uses of the Source game engine in psychological research. Furthermore, we have demonstrated several approaches to summarizing the large amount of data collected by this methodology. With our data extraction utility we were able to create distance and time indices that were sensitive to the different approaches taken by the two performance groups and have provided an example of how to represent and interpret such data. It is evident that alternative approaches, such as post-test video coding, could not easily produce the level of movement detail required for the distance analysis we report (for examples of the use of the DEU for other data such as participant actions, action sequences, and action timing, and how to rep-

resent the movement data graphically see Logie et al., 2011; Trawley et al., 2011). Another important consideration for researchers is how future-proof is the methodology: whether a software program could be outdated and unsupported in a year's time is an important factor in deciding whether to commit resources to it. However, such concerns are alleviated by the continued support of the Steam engine and regular updates for the SDK (Source SK release notes archive). Moreover, the new facility that we have provided will allow researchers to set up and extract data from VR experiments very rapidly, allowing substantial research to be completed over a short development time scale. While potential future advances in VR software and technology that are inexpensive and incompatible with our methodology are possible, this solution still offers substantial advantages for the researcher.

One may argue that a fully-immersive virtual environment (commonly referred as a CAVE; computer automatic virtual environment) provides better ecological validity than desktop VR, with the same ability to retain experimental control and collect and summarize detailed behavioral data. However, there are important reasons why researchers might choose the desktop approach, the most obvious being the lower financial cost. For example, Green and White (2000) estimated \$100,000 USD for a low-cost CAVE system. Even for a CAVE implementation using a budget game engine the cost is very large compared with the methodology that we are proposing (see Jacobson & Lewis, 2005; Juarez, Schonenberg, & Bartneck, 2010, who both estimated this approach would cost approximately \$25,000 USD). The approach described in this paper provides a VR solution that compares favorably to the cost of entry of these approaches. In addition, some participant groups (e.g., older adults) might find fully-immersive VR confusing and disorienting, which would undermine the aim of studying their behavior in a naturalistic setting. In contrast, the use of a single desktop monitor will be familiar to most participants as it has strong similarities with a standard office setup. Furthermore, work has shown that older participants are at a higher risk of simulator sickness (see Classen, Bewernitz, & Shechtman, 2011; Brooks et al., 2010), and use of non-immersive VR has been shown to be less likely to increase feelings of nausea than fully immersive systems (Costello & Howarth, 1996).

In summary, we have successfully developed and documented a low cost, accessible and flexible procedure for running desktop VR studies for researchers with no programming experience. Work by our group has shown that the data collected by this approach is an effective strategy for assessing specific hypotheses (Logie et al., 2011; Trawley et al., 2011). By detailing the methodology and providing accessible data extraction software to interested researchers we hope to encourage the use of this flexible technique for a variety of research topics in cognition.

References

- Ayaz, H., Allen, S. L., Platek, S. M., & Onaral, B. (2008). Maze Suite 1.0: a complete set of tools to prepare, present, and analyze navigational and spatial cognitive neuroscience experiments. *Behavior research methods*, *40*(1), 353–359. doi: 10.3758/BRM.40.1.353
- Becker-Asano, C., Sun, D., Kleim, B., Scheel, C. N., Tuschen-Caffier, B., & Nebel, B. (2011). Outline of an empirical study on the effects of emotions on strategic behavior in virtual emergencies. In *Affective computing and intelligent interaction* (pp. 508–517). Springer. doi: 10.1007/978-3-642-24571-8_64
- Bodily, K. D., Daniel, T. A., & Sturz, B. R. (2012). The roles of beaconing and dead reckoning in human virtual navigation. *Learning and Motivation*, *43*(1), 14–23. doi: 10.1016/j.lmot.2012.01.002
- Bodily, K. D., Eastman, C. K., & Sturz, B. R. (2011). Neither by global nor local cues alone: evidence for a unified orientation process. *Animal cognition*, *14*(5), 665–674. doi: 10.1007/s10071-011-0401-x
- Brooks, J. O., Goodenough, R. R., Crisler, M. C., Klein, N. D., Alley, R. L., Koon, B. L., . . . Wills, R. F. (2010). Simulator sickness during driving simulation studies. *Accident Analysis & Prevention*, *42*(3), 788–796. doi: 10.1016/j.aap.2009.04.013
- Cenkner, A., Bulitko, V., & Spetch, M. (2011). A generative computational model for human hide and seek behavior. In *Proceedings of the seventh artificial intelligence and interactive digital entertainment conference (aiide 2011)*. palo alto, ca: Aaai press (pp. 128–133).
- Classen, S., Bewernitz, M., & Shechtman, O. (2011). Driving simulator sickness: an evidence-based review of the literature. *The American Journal of Occupational Therapy*, *65*(2), 179–188. doi: 10.5014/ajot.2011.000802
- Costello, P., & Howarth, P. (1996). The visual effects of immersion in four virtual environments. *Produced under HSE Research Contract, 3181*, R53.
- Dyck, M., Winbeck, M., Leiberg, S., Chen, Y., & Mathiak, K. (2010). Virtual faces as a tool to study emotion recognition deficits in schizophrenia. *Psychiatry research*, *179*(3), 247–252. doi: 10.1016/j.psychres.2009.11.004
- Feng, J., Spence, I., & Pratt, J. (2007). Playing an action video game reduces gender differences in spatial cognition. *Psychological Science*, *18*(10), 850–855. doi: 10.1111/j.1467-9280.2007.01990.x
- Frey, A., Hartig, J., Ketzler, A., Zinkernagel, A., & Moosbrugger, H. (2007). The use of virtual environments based on a modification of the computer game Quake III Arena® in psychological experimenting. *Computers in Human Behavior*, *23*(4), 2026–2039. doi: 10.1016/j.chb.2006.02.010
- Gatzidis, C., Brujic-Okretic, V., & Baker, S. (2008, July). A review of first-person shooter game engines and their use in researching scientific disciplines. In *IADIS international conference gaming 2008: Design for engaging experience and social interaction* (pp. 67–76). Amsterdam, Netherlands: IADIS Press.
- Glotzbach-Schoon, E., Tadda, R., Andreatta, M., Tröger, C., Ewald, H., Grillon, C., . . . Mühlberger, A. (2013). Enhanced discrimination between threatening and safe contexts in high-anxious individuals. *Biological psychology*. doi: 10.1016/j.biopsycho.2013.01.011
- Green, M., & White, L. (2000). The cave-let: a low-cost projective immersive display. *Journal of telemedicine and telecare*, *6*(suppl 2), 24–26. doi: 10.1258/1357633001935437
- Greville, W. J., Newton, P. M., Roche, B., & Dymond, S. (2013). Conditioned suppression in a virtual environment. *Computers in Human Behavior*, *29*(3), 552–558. doi: 10.1016/j.chb.2012.11.016
- Hartley, T., Trinkler, I., & Burgess, N. (2004). Geometric determinants of human spatial memory. *Cognition*, *94*(1), 39–75. doi: 10.1016/j.cognition.2003.12.001
- Hartmann, T., & Vorderer, P. (2010). It's okay to shoot a character: Moral disengagement in violent video games. *Journal of Communication*, *60*(1), 94–119. doi: 10.1111/j.1460-2466.2009.01459.x
- Jacobson, J., & Lewis, M. (2005). Game engine virtual reality with caveat. *Computer*, *38*(4), 79–82. doi: 10.1109/MC.2005.126
- Jeong, E. J., Biocca, F. A., & Bohil, C. J. (2012). Sensory realism and mediated aggression in video games. *Computers in Human Behavior*. doi: 10.1016/j.chb.2012.05.002
- Jouriles, E. N., McDonald, R., Kullowatz, A., Rosenfield, D., Gomez, G. S., & Cuevas, A. (2009). Can virtual reality increase the realism of role plays used to teach college women sexual coercion and rape-resistance skills? *Behavior therapy*, *40*(4), 337–345. doi: 10.1016/j.beth.2008.09.002
- Juarez, A., Schonenberg, W., & Bartneck, C. (2010). Implementing a low-cost cave system using the cryengine2. *Entertainment Computing*, *1*(3), 157–164. doi: 10.1016/j.entcom.2010.10.001
- Kivikangas, M. J., Nacke, L., & Ravaja, N. (2011). Developing a triangulation system for digital game events, observational video, and psychophysiological data to study emotional responses to a virtual character. *Entertainment Computing*, *2*(1), 11–16. doi: 10.1016/j.entcom.2011.03.006
- Kozlov, M. D., & Johansen, M. K. (2010). Real behavior in virtual environments: Psychology experiments in a simple virtual-reality paradigm using video games. *Cyberpsychology, behavior, and social networking*,

- 13(6), 711–714. doi: 10.1089/cyber.2009.0310
- Krotoski, A. (2009). *Social influence in second life: Social network and social psychological processes in the diffusion of belief and behavior on the web*. Unpublished doctoral dissertation, University of Surrey, Guildford, United Kingdom.
- Law, A. S., Logie, R. H., & Pearson, D. G. (2006). The impact of secondary tasks on multitasking in a virtual environment. *Acta psychologica*, 122(1), 27–44. doi: 10.1016/j.actpsy.2005.09.002
- Law, A. S., Trawley, S. L., Brown, L. A., Stephens, A. N., & Logie, R. H. (2012). The impact of working memory load on task execution and online plan adjustment during multitasking in a virtual environment. *The Quarterly Journal of Experimental Psychology*(ahead-of-print), 1–18. doi: 10.1080/17470218.2012.748813
- Legge, E. L., Madan, C. R., Ng, E. T., & Caplan, J. B. (2012). Building a memory palace in minutes: Equivalent memory performance using virtual versus conventional environments with the method of loci. *Acta psychologica*, 141(3), 380–390. doi: 10.1016/j.actpsy.2012.09.002
- Legge, E. L., Spetch, M. L., Cenkner, A., Bulitko, V., Anderson, C., Brown, M., & Heth, D. (2012). Not all locations are created equal: Exploring how adults hide and search for objects. *PloS one*, 7(5), e36993. doi: 10.1371/journal.pone.0036993.g013
- Logie, R. H., Trawley, S. L., & Law, A. S. (2011). Multitasking: multiple, domain-specific cognitive functions in a virtual environment. *Memory & cognition*, 39(8), 1561–1574. doi: 10.3758/s13421-011-0120-1
- McGeorge, P., Phillips, L. H., Crawford, J. R., Garden, S. E., Sala, S. D., Milne, A. B., ... Callender, J. S. (2001). Using virtual environments in the assessment of executive dysfunction. *Presence: Teleoperators & Virtual Environments*, 10(4), 375–383. doi: 10.1162/1054746011470235
- McQuiggan, S. W., Robison, J. L., Phillips, R., & Lester, J. C. (2008). Modeling parallel and reactive empathy in virtual agents: An inductive approach. In *Proceedings of the 7th international joint conference on autonomous agents and multiagent systems-volume 1* (pp. 167–174).
- Miller, J., & Carlson, L. (2011). Selecting landmarks in novel environments. *Psychonomic bulletin & review*, 18(1), 184–191. doi: 10.3758/s13423-010-0038-9
- Nacke, L. E., Grimshaw, M. N., & Lindley, C. A. (2010). More than a feeling: Measurement of sonic user experience and psychophysiology in a first-person shooter game. *Interacting with Computers*, 22(5), 336–343. doi: 10.1016/j.intcom.2010.04.005
- Nacke, L. E., Lindley, C., & Stellmach, S. (2008). Log who's playing: psychophysiological game analysis made easy through event logging. In *Fun and games* (pp. 150–157). Springer. doi: 10.1007/978-3-540-88322-7_1
- Radvansky, G. A., & Copeland, D. E. (2006). Walking through doorways causes forgetting: Situation models and experienced space. *Memory & Cognition*, 34(5), 1150–1156. doi: 10.3758/BF03193261
- Radvansky, G. A., Tamplin, A. K., & Krawietz, S. A. (2010). Walking through doorways causes forgetting: Environmental integration. *Psychonomic bulletin & review*, 17(6), 900–904. doi: 10.3758/PBR.17.6.900
- Smith, S. P., & Du'Mont, S. (2009). Measuring the effect of gaming experience on virtual environment navigation tasks. In *3d user interfaces, 2009. 3dUI 2009. IEEE symposium on* (pp. 3–10).
- Smith, S. P., & Trenholme, D. (2009). Rapid prototyping a virtual fire drill environment using computer game technology. *Fire safety journal*, 44(4), 559–569. doi: 10.1016/j.firesaf.2008.11.004
- Sturz, B. R., & Bodily, K. D. (2010). Encoding of variability of landmark-based spatial information. *Psychological research*, 74(6), 560–567. doi: 10.1007/s00426-010-0277-4
- Sturz, B. R., & Bodily, K. D. (2011). Is surface-based orientation influenced by a proportional relationship of shape parameters? *Psychonomic bulletin & review*, 18(5), 848–854. doi: 10.3758/s13423-011-0111-z
- Sturz, B. R., Bodily, K. D., & Katz, J. S. (2009). Dissociation of past and present experience in problem solving using a virtual environment. *CyberPsychology & Behavior*, 12(1), 15–19. doi: 10.1089/cpb.2008.0147
- Sturz, B. R., & Brown, M. F. (2009). Facilitation of learning spatial relations among locations by visual cues: Implications for theoretical accounts of spatial learning. *Psychonomic bulletin & review*, 16(2), 306–312. doi: 10.3758/PBR.16.2.306
- Sturz, B. R., & Diemer, S. M. (2010). Reorienting when cues conflict: A role for information content in spatial learning? *Behavioural processes*, 83(1), 90–98. doi: 10.1016/j.beproc.2009.11.001
- Sturz, B. R., Forloines, M. R., & Bodily, K. D. (2012). Enclosure size and the use of local and global geometric cues for reorientation. *Psychonomic bulletin & review*, 19(2), 270–276. doi: 10.3758/s13423-011-0195-5
- Sturz, B. R., Gaskin, K. A., & Bodily, K. D. (2013). Overtraining and the use of feature and geometric cues for reorientation. *Psychological research*, 77(2), 176–182. doi: 10.1007/s00426-011-0410-z
- Sturz, B. R., Gurley, T., & Bodily, K. D. (2011). Orientation in trapezoid-shaped enclosures: implications for theoretical accounts of geometry learning. *Journal of Experimental Psychology-Animal Behavior Processes*, 37(2), 246. doi: 10.1037/a0021215

- Sturz, B. R., & Kelly, D. M. (2009). Encoding of relative enclosure size in a dynamic three-dimensional virtual environment by humans. *Behavioural processes*, *82*(2), 223–227. doi: 10.1016/j.beproc.2009.06.007
- Sturz, B. R., Kelly, D. M., & Brown, M. F. (2010). Facilitation of learning spatial relations among locations by visual cues: Generality across spatial configurations. *Animal cognition*, *13*(2), 341–349. doi: 10.1007/s10071-009-0283-3
- Sturz, B. R., Paul Cooke, S., & Bodily, K. D. (2011). Solving for two unknowns: An extension of vector-based models of landmark-based navigation. *Journal of Experimental Psychology-Animal Behavior Processes*, *37*(3), 368. doi: 10.1037/a0022938
- Sutton, J. E., Joannis, M. F., & Newcombe, N. S. (2010). Spinning in the scanner: Neural correlates of virtual reorientation. *Journal of experimental psychology. Learning, memory, and cognition*, *36*(5), 1097. doi: 10.1037/a0019938
- Talbot, K. J., Legge, E. L., Bultko, V., & Spetch, M. L. (2009). Hiding and searching strategies of adult humans in a virtual and a real-space room. *Learning and Motivation*, *40*(2), 221–233. doi: 10.1016/j.lmot.2009.01.003
- Tlauka, M., Carter, P., Mahlberg, T., & Wilson, P. N. (2011). The first-perspective alignment effect: The role of environmental complexity and familiarity with surroundings. *The Quarterly Journal of Experimental Psychology*, *64*(11), 2236–2250. doi: 10.1080/17470218.2011.586710
- Trawley, S. L., Law, A. S., & Logie, R. H. (2011). Event-based prospective remembering in a virtual world. *The Quarterly Journal of Experimental Psychology*, *64*(11), 2181–2193. doi: 10.1080/17470218.2011.584976
- Walshe, D. G., Lewis, E. J., Kim, S. I., O’Sullivan, K., & Wiederhold, B. K. (2003). Exploring the use of computer games and virtual reality in exposure therapy for fear of driving following a motor vehicle accident. *CyberPsychology & Behavior*, *6*(3), 329–334. doi: 10.1089/109493103322011641
- Washburn, D. A. (2003). The games psychologists play (and the data they provide). *Behavior Research Methods, Instruments, & Computers*, *35*(2), 185–193. doi: 10.3758/BF03202541