Low-cost optical tracking for immersive collaboration in the CAVE using the Wii Remote

A Murgia[1], R Wolff[2], P M Sharkey[1] and B Clark[1]

[1]School of Systems Engineering, University of Reading,
Whiteknights, Reading, UK

[2]The Centre for Virtual Environments, University of Salford,
University Road, Salford, UK

a.murgia@reading.ac.uk, *r.wolff@salford.ac.uk*, *p.m.sharkey@reading.ac.uk*

ABSTRACT

We present a novel way of interacting with an immersive virtual environment which involves inexpensive motion-capture using the Wii Remote®. A software framework is also presented to visualize and share this information across two remote CAVE™-like environments. The resulting applications can be used to assist rehabilitation by sending motion information across remote sites. The application's software and hardware components are scalable enough to be used on desktop computer when home-based rehabilitation is preferred.

1. INTRODUCTION

In the last decade, the use of virtual reality (VR) to assist therapy has received increasing support from the clinical community (Moline, 1997). Advantages provided by virtual reality platforms include ecological validity, that is relevance to real world applications, stimulus consistency, and an easy way to provide motivation by user-friendly and graphically-attractive environments. However at present technological developments limit the interaction between user and environment, and make the process of moving in the environment not comfortable (Rizzo and Kim, 2005). Virtual reality investments also suffer from higher costs of equipment. This type of investment is not appealing to health professionals unless substantial benefits can be projected in return. Thus the recent focus of engineers working in VR has been to scale the technology down so that it is computational lightweight and physically portable, while normal prototyping still takes place on a more expensive scale.

Major applications that currently use VR include, among others, stroke rehabilitation (August et al., 2005; Dobkin, 2004; Merians et al., 2002), and helping locomotion in Parkinson disease (Sveistrup, 2004). In particular, upper-limb stroke therapy has benefitted from neurological research that points to movement exercise as a fundamental tool to recover motion by exploiting neural plasticity (Krakauer, 2005; Nudo and Friel, 1999; Taub et al., 1993; Winstein et al., 1999). This has led to detailed studies of upper limb kinematics in stroke subjects in order to understand motor response after stroke, as well as to design more effective therapies (Levin, 1996; Micera et al., 2005). Virtual reality applications have emerged that target stroke treatment and provide assistance to the therapists during motion exercises. They include assisting movements by robotic mechanisms (Loureiro et al., 2003), capturing hand range of motion (August et al., 2005), speed and finger force using data gloves, as well as providing feedback on exercise performance (Merians et al., 2002). Other applications use a web-based library that is queried by the user during therapy to provide exercises and feedback on performance (Reinkensmeyer et al., 2002). Interactions with the environment happen via a mouse or a low-cost force-feedback joystick or haptic glove (Popescu et al., 2000). These applications provide motivation through the use of interactive environments, puzzles and games. Commercial game platforms (EyeToy™) have also been used to monitor 2D upper limb movements and provide a rehabilitation treatment (Rand et al., 2004). However, in order to capture 3D movements, extended to different upper limb segments, more expensive systems need to be used, including optical or magnetic motion tracking. These technologies come at comparatively high costs (> US$ 20K) for the health service, which in turn means that they are also constrained by the location where they are used.

We present a pilot application where the wireless remote controller Wii Remote® (Wiimote) of the Nintendo Wii® game console is used to perform motion-capture in immersive collaborative virtual reality (VR) environments. The application can also run on non-immersive platforms and desktop computers for a more widespread distribution. The Wiimote provides an inexpensive alternative for 3D optical motion-capture (~ US$ 60 / Wiimote). It integrates, among others, an in-built infrared (IR) camera with on-chip processing, accelerometers and supports Bluetooth® communication. This characteristic makes it suitable to communicate with external hardware that supports Bluetooth, while several open-source libraries are available to capture and process the derived information. The display environments used in the pilot trials presented here are two CAVE™-like systems, which allow perception of 3D stereo graphics from a first-person viewpoint, and have the advantage of an enhanced sense of presence and increased interaction among participants. These were linked via the Internet using a collaborative virtual environment (CVE) software system that runs as a distributed simulation. Alternatively the software can be ported on desktop computers or single screen projection systems without stereo visualization.

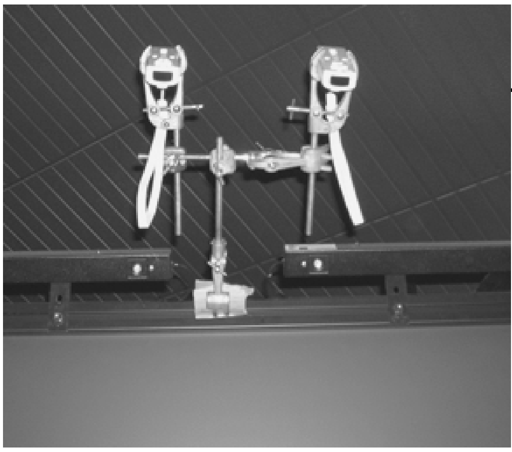
Here, information derived from the Wiimote camera is used to track an IR source, mounted for example on a portable support, and to generate graphics corresponding to the movements of the user. These data are successively shared across the networked CAVE displays via the CVE application.

2. METHODS

When used in the game console the Wiimote tracks the position of two fixed IR sources mounted externally on a device known as 'sensor bar' using the in-built infrared camera. The IR sources appear to the camera as bright spots on a dark background. From these 2D images, the 3D position of the Wiimote is calculated based on the known distance between the 2 fixed IR sources in the sensor bar.

In our applications, the Wiimote was used in the reversed way. Two Wiimotes were mounted on fixed supports inside each CAVE, while the participants moved an IR source. The IR source (wavelength 950 nm) was captured by both Wiimote cameras and this information was communicated to a computer enabled to receive Bluetooth and which run the server software (referred to as wiitrack). The IR source was easily created from an IR LED (IR pen), or alternatively by shining IR light generated via a LED array on a reflective marker attached to the participant's hand. Figure 1 shows the physical mounting of the Wiimotes in one CAVE.

The 3D coordinates of the IR source were calculated locally by each wiitrack application and sent to each local CAVE-rendering application (CVEclient), while the 3D coordinates of the IR source in the remote CAVE were communicated across the 2 sites via relay server application (CVEserver). A scheme of the data flow and software is shown in Figure 1. The system was tested by 2 healthy participants, working at each remote site in a series of accuracy tests and pilot rehabilitation trials.



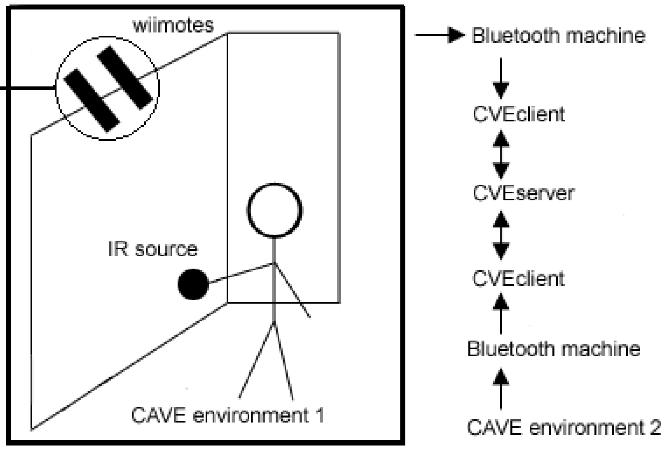


Figure 1. *Left: Physical mounting of the Wiimotes on top of the CAVE front wall and IR pen. Right: Wiimote use in a shared virtual reality environment. CVE: Collaborative Virtual Environment.*

2.1 Wiitrack Server

Each Wiimote has an IR camera with a 1024x768 pixels resolution and approximately 40 degrees field of view. The 2D coordinates of the IR source as seen by each Wiimote camera were read by the wiitrack application using the wiiuse (Laforest, 2008) open-source library. These 2D coordinates were then filtered independently for each Wiimote using a Butterworth filter with a ratio between sampling over cutoff frequency: $f_s/fc = 8$. Sampling rate of each Wiimote camera was was 100 Hz, transmission rate between wiitrack and CVEclient was 30 ms. The three-dimensional position of the IR source was calculated using stereo triangulation with 2 parallel Wiimotes facing the IR source (see Figure 2) as:

$$\begin{aligned} X_{actual} &= x_{cleft} \frac{Z_{actual}}{f} \\ Y_{actual} &= y_{cleft} \frac{Z_{actual}}{f} \\ Z_{actual} &= \frac{b \cdot f}{x_{cleft} - x_{cright}} \end{aligned} \quad . \qquad (1)$$

Where the Z axis is the direction towards which the cameras are pointing, b is the distance between the cameras, f is the focal length, *xcleft* and *xcright* are x IR source coordinates on the left and right view planes respectively.

A graphical user interface (GUI) was written for wiitrack in C++ to take into account parameters such as Wiimotes origin location, rotation and Wiimote camera sensitivity. The latter is adjustable from a level 1 to a maximum of 5 in wiiuse. Averaging the past n positions to calculate the $n^{th}+1$ position was also included as an option, both before and after filtering took place.

The reason for implementing the tracker algorithm in a separate application, rather than within the CVEclient itself, was that the visualisation computer driving the CAVE did not support Bluetooth natively. Therefore, wiitrack had to run on a separate computer and communicate the tracking data via the local area network (LAN) to the CVEclient. However, this server-based tracker solution also allows other applications than the CVEclient to access the tracking data virtually in parallel, e.g. to log a patient's performance for later analysis. Communication between wiitrack server and CVEclient was implemented using the open-source library RakNet (Jenkins_Software, 2008). Another open-source option also available is the VRPN (Taylor II et al., 2001). The network delay for delivering tracking data from the wiitrack server to the CVEclient was typically between 16 to 19 ms.

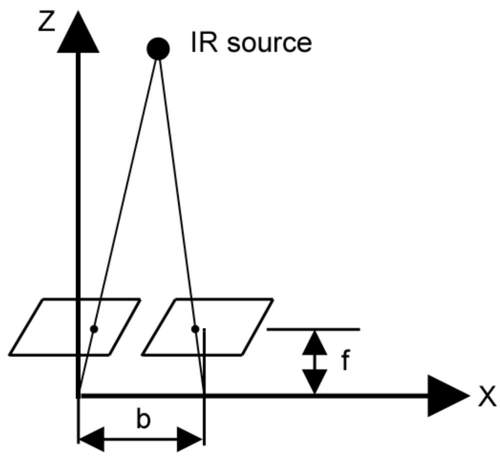


Figure 2. *Stereovision for 2 Wiimotes. The Wiimotes axes are parallel to the Z axis. The 2 parallelograms represent the view planes of the Wiimotes, perpendicular to the Z axis (Society_of_Robots, 2008).*

2.2 CVEclient and CVEserver

The CVEclient was used to visualise the shared virtual world. Rendering was done using the open-source library OpenSG (Reiners and Voss, 2008). Grabbing and moving virtual objects, as well as navigating through the scene were supported. Motion tracking of the user's head and hand allowed interaction with the virtual world. A person connected via a remote CAVE was represented by a computer-generated character, referred to as avatar. The realism of an avatar's appearance was increased by mapping photographs of real humans onto the 3D polygonal mesh which outlined the avatar's model. Dynamic head and arm articulation has been enabled by providing separate body parts for the head, hand, upper- and lower-arm parts. The head and hand coordinates were controlled directly from the tracker data. Animation of the remaining arm parts was implemented by applying a simple inverse kinematic algorithm. During the experiments, we used the original head tracking, while we swapped the hand tracking with our Wiimote-based implementation.

The CVEserver had two functions. Firstly, it acted as login server, similar to a directory server, where a client can chose which virtual world to join. Secondly, it acted as relay server, passing all communication between CVEclients that joined the same virtual world in order to synchronise their shared state. This could have been solved on a peer-to-peer basis in a more scalable manner. However, using a relay server simplifies firewall administration and moves the overhead for broadcasting network messages to all peers connected to the server.

Typical measured network delay between the CVEclients ranged between 36 to 42 ms, while the actual end-to-end latency was around 120 to 160 ms. The end-to-end latency includes the time it takes, from the moment the tracker's data are acquired, to transmit these data to the CVEclient, pass them via the CVEserver to the receiving CVEclient, apply them to the virtual scene, until they are finally rendered on the screen. An audio link-up between the remote users was provided using Skype®.

2.3 Measurement of Accuracy

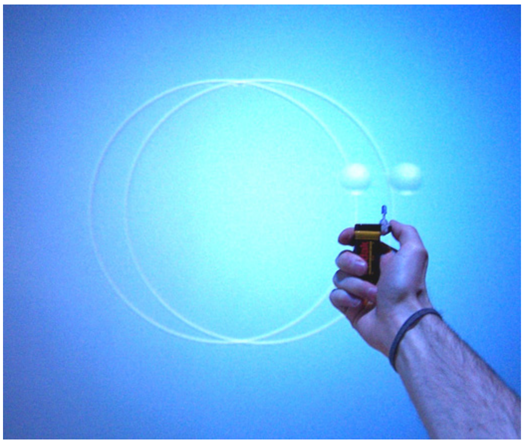
Measurement of accuracy was carried out by comparing the errors generated while one participant followed a pre-defined trajectory using either the IR pen or the CAVE standard wand (used as reference). The trajectory to follow was made visible to the participant as a solid closed line, which the participant was asked to trace in a way similar to a wire loop game, repeating the process 5 times. A 3D pointer in front of the IR pen, or the wand, allowed to receive feedback on the pointer position. The errors between tracked and actual positions were calculated and normalized over the actual position in order to have a normalized error percentage:

$$E_{norm} = \frac{D_{tracked} - D_{actual}}{D_{actual}}. \tag{2}$$

Where $D_{tracked}$ is the distance from the origin tracked using either the IR pen or the wand and D_{actual} is the actual position where the tracked point should have been, i.e. the loop's radius. The E_{norm} distribution while using the wand was compared at a 5% significance level with the E_{norm} distribution while using the IR pen in 6 different settings: filters on or off together with number of previous positions averaged set to 1 (no average), 10 and 20: $IR_{on,\,1,\,10,\,20}$; $IR_{off,\,1,\,10,\,20}$.

2.4 Relevance to Rehabilitation

Among the different applications which were considered, including 3D drawing, we decided to implement a rehabilitation scenario. In this application, which targets upper limb rehabilitation, one participant simulated the person receiving therapy (referred to as local user), moving the IR source to interact in the virtual space as instructed by a remotely located therapist. The actions and movements of the local user were also available to the therapist in real time. One scenario provided the local user with a set of objects located at different reach lengths. The local user was instructed to pick and place these objects in a close-by container. A screenshot from a recorded virtual session is shown in Figure 3, while the CAVE environment at the 'therapist' site is shown in Figure 4. Locking and unlocking the object's position (pick and place) with the IR pointer was controlled by the local user by pressing a button on the wand, but was only allowed to happen when the IR-controlled pointer was moved by the local user so that it intersected the object's volume. Different exercises can thus be designed to cover specific range of movements of the upper limbs by placing objects at different distances or asking the local user to trace specific paths with varying level of difficulty.



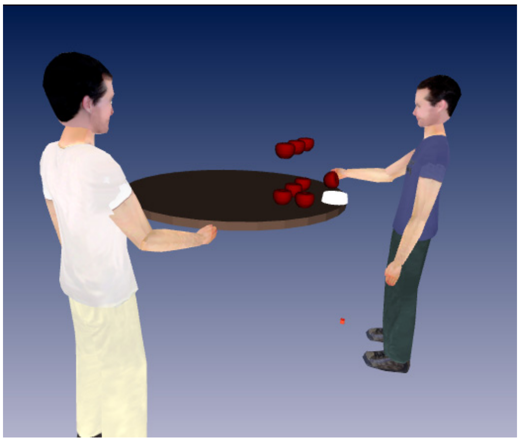


Figure 3. *Left: Accuracy tests with IR pen. Right: Rehabilitation scenario. The participant playing the patient role was instructed to pick each apple in turn and place it in a bowl.*

3. RESULTS

The pilot trials showed that IR source did not appear to interfere greatly with the IR emitters that synchronize the CAVE shutter glasses, consequently perception of the 3D scene was not disrupted. It was noticed however that the IR pointer was noisier than the equivalent wand-controlled pointer when the number of previous points averaged was less than 10. For an averaged number of points equal to 10 or above, the movement of the IR-controlled pointer was smooth, although the lag between the actual position of the hand and that of the 3D pointer increased. A value of 20 average points introduced a lag that was still comfortable to cope with at slow speed but was noticeable for reasonably quicker hand movements (speed > 1 m/s).

Variances between the E_{norm} distribution using the wand and the IR pen were different in all cases except for $IR_{on,\ 20}$ (n_1 = 1627 samples, n_2 = 1179 samples; P = 0.193). Means between the wand and IR pen populations were different in all IR settings ($P \leq 0.01$). The analysis also reveled that the variances for $IR_{on\ 1}$ (P = 0), $IR_{on\ 10}$ (P < 1E-5), and $IR_{off\ 1,\ 10,\ 20}$ (P = 0) were statistically less than that of the wand population.

As the Wiimote base was not movable, the capturing volume, i.e. the volume inside which an IR source would be visible by both Wiimotes, was dependent on the Wiimotes field of view. For Wiimotes mounted 19 cm apart on one of the cave walls (total height about 2.4 m), an approximate volume of less than 0.5 m^3, centered in the middle of the CAVE volume, was visible by both devices. This visible volume allowed relative easy movement during the accuracy tests (loop diameter was 30 cm approximately) and during pilot tests in the rehabilitation scenario.

Finally, the position of the IR pointer in the virtual environment depended on the accurate specification of the Wiimotes origin in the GUI. Several attempts were required during calibration to identify the correct parameters that reflected the movement of the real IR source on the virtual 3D pointer.

4. DISCUSSION

The statistic analysis revealed that the error variability while tracking a loop figure was significantly less for the IR populations than for the wand in most cases. However these results have to be interpreted carefully as it is possible that, when using the wand, the position of the virtual 3D pointer as well as its size in respect to the real wand may have influenced the variability of the wand population. For example if the wand is perceived to be too big relatively to the virtual pointer, or the pointer too far away, the user may be induced to make several positional adjustments compared to a case when it is not.

When using the IR source, increasing noise when none or less than 10 points averaging was used, could be explained by considering that the resolution of the CAVE wall and that of the Wiimote camera are the same (1024x768 pixels). However the dimensions of the CAVE projection wall are 3 m x 2.2 m compared to few millimeters on the camera. This means that a small movement seen on the camera's viewing plane can be translated into a relatively larger movement on the CAVE wall, hence the noisier response.

The attempts to specify the correct parameters that identify the Wiimotes location can cause unwanted delays. For this reason a different way of calibrating the system will be sought in future applications. This includes moving an array of IR sources (for example LEDs), whose geometrical arrangement is known beforehand, in front of the Wiimotes in a way similar to that used by some commercial motion capture systems. During the pilot tests it was found that the capturing volume available allowed tracking the full range of motion of the elbow while the shoulder was flexed to reach objects placed in front of the subject. Movements involving motions of the shoulder in the frontal plane, or involving subjects walking around the CAVE can also be tracked provided the Wiimotes support rotates about its vertical axis. Although this solution would increase the cost of the tracking mechanism, it is reasonably easy to implement using a single motor that is controlled to compensate for the error between the IR source on the Wiimote's view plane and the plane's centre.

Tele-rehabilitation can benefit from inexpensive motion capture as the movements of somebody receiving therapy can be sent to a therapist located in a different site in real time at a reasonably low cost. In this pilot application we tracked a single IR source; however multiple IR sources can also be used. The advantage of tracking multiple IR sources is that joint movements can be calculated if the sources are attached to specific bone landmarks. The 3D coordinates of multiple sources can be calculated from 2 Wiimotes as long as the sources are easy to distinguish on the Wiimote's view plane. Multiple Wiimotes can be used when this is no longer possible. This can be done at low cost if the sources are no longer active but passive reflectors. The alternative to using an IR pen is to use a passive reflective marker and an IR lamp around every IR camera. A solution is currently being built that allows this.

The immersive virtual environments we use become impractical and expensive if remote rehabilitation of a large number of individuals is envisaged. However, the proposed technological framework has the advantage of being easily scalable to allow home-based rehabilitation. In this scenario the server and client software would reside on desktop PCs. The therapist would connect remotely from his hospital-based PC to the patient's home PC, which is equipped with the motion-capture hardware described above. An interactive rehabilitation program can thus be created at a relatively low cost while reducing the patient's need to travel. To improve this rehabilitation system, it is possible to provide a set of feedback measurements. These include the normalized error cited above, the range of motion of selected joints, e.g. the elbow, and the total length traveled by the hand segment. These feedback measures can be conveyed by the therapist in real time via the audio link to stimulate patient motivation and involvement.

Figure 4. *View of the remotely-connected CAVE and the participant playing the therapist role. The objects' projections are visible on the front screen.*

5. CONCLUSIONS

We proposed an inexpensive motion-capture solution integrated into an immersive virtual environment which has the potential to assist remote movement-based rehabilitation. The proposed motion-capture solution has the advantage that hardware parts are relatively inexpensive compared to a commercial motion capture system, since only a minimum of two Wiimotes and an IR source are needed to perform triangulation in 3D. The application link-up can allow a therapist in one location to look at the motions of a patient in a remote location and to instruct interactively via an audio link already present in the CAVE systems. The therapist can also provide information related to the patient performance, thus increasing participation and motivation. Software scalability and lightweight tracker hardware make the system portable to desktop applications and suitable for home-based rehabilitation.

6. REFERENCES

K August, D Bleichenbacher and S Adamovich (2005), Virtual reality physical therapy: a telerehabilitation tool for hand and finger movement exercise monitoring and motor skills analysis, Proc. IEEE 31st Annual Northeast Bioengineering Conference, Hoboken, NJ.

B Dobkin (2004), Strategies for stroke rehabilitation, The Lancet, 3, pp. 528-536.

Jenkins_Software (2008), http://www.jenkinssoftware.com/.

J W Krakauer (2005), Arm function after stroke: From physiology to recovery, Seminars in Neurology, 25, pp. 384-395.

M Laforest (2008), http://www.wiiuse.net/.

M F Levin (1996), Interjoint coordination during pointing movements is disrupted in spastic hemiparesis, Brain, 119, pp. 281-293.

R C V Loureiro, F Amirabdollahian, M Topping, B Driessen and W S Harwin (2003), Upper limb robot mediated stroke therapy - GENTLE/s approach, Autonomous Robots, 15, pp. 35-51.

A S Merians, D Jack, R Boian, M Tremaine, G C Burdea, S V Adamovich, M Recce and H Poizner (2002), Virtual reality–augmented rehabilitation for patients following stroke, Physical Therapy, 82, pp. 898-915.

S Micera, J Carpaneto, F Posteraro, L Cenciotti, M Popovic and P Dario (2005), Characterization of upper arm synergies during reaching tasks in able-bodied and hemiparetic subjects, Clinical Biomechanics, 20, pp. 939-946.

J Moline (1997), Virtual reality for health care: a survey, Studies in Health Technology and Informatics, 44, pp. 3-34.

R J Nudo and K M Friel (1999), Cortical plasticity after stroke: implications for rehabilitation, Revue Neurologique, 155, pp. 713-717.

V G Popescu, G C Burdea, M Bouzit and V R Hentz (2000), A virtual-reality-based telerehabilitation system with force feedback, IEEE Transactions on Information Technology in Biomedicine, 4.

D Rand, R Kizony and P L Weiss (2004), Virtual reality rehabilitation for all: Vivid GX versus Sony PlayStation II EyeToy, Proc. 5th International Conference On Disability, Virtual Reality And Associated Technologies, Oxford, UK.

D Reiners and G Voss (2008), http://opensg.vrsource.org/trac.

D J Reinkensmeyer, C T Pang, J A Nessler and C C Painter (2002), Web-based telerehabilitation for the upper extremity after stroke, IEEE Transactions on Neural Systems and Rehabilitation Engineering, 10, pp. 102-108.

A Rizzo and G J Kim (2005), A SWOT analysis of the field of virtual reality rehabilitation and therapy, Presence: Teleoperators and Virtual Environments, 14, pp. 119-146.

Society_of_Robots (2008), http://www.societyofrobots.com/programming_computer_vision_tutorial_pt3.shtml.

H Sveistrup (2004), Motor rehabilitation using virtual reality, Journal of NeuroEngineering and Rehabilitation, 1:10.

E Taub, N E Miller, T A Novack, E W Cook, W C Fleming, C S Nepomuceno, J S Connell and J E Crago (1993), Technique to improve chronic motor deficit after stroke, Archives of Physical Medicine and Rehabilitation, 74, pp. 347-354.

R M Taylor II, T C Hudson, A Seeger and H Weber (2001), VRPN: A device-independent, network-transparent VR peripheral system, Proc. ACM Symposium on Virtual Reality Software & Technology 2001, Banff, Canada.

C J Winstein, A S Merians and K J Sullivan (1999), Motor learning after unilateral brain damage, Neuropsychologia, 37, pp. 975-987.