

Presented by: Stephen Witherden
Senior Software Developer
Beca Applied Technologies
stephen.witherden@beca.com

Abstract. The Kinect sensor from Microsoft presents a uniquely affordable approach to facilitate sophisticated interaction with virtual environments. When applied to serious gaming environments such as VBS2, this technology can be used to train trainees in movements which concern their whole bodies. In this case, we have focused our attention on recognising and responding to the gestures required for the task of Air Marshalling. Use of this technology enables the trainee to learn, practise and retain the muscle memory involved in performing the air marshalling signals.

The virtual environment can be configured for virtually any surface or air platform. In addition, dangerous situations such as hydraulic or engine failures can be simulated without risk to the air and ground crew. It is expected to be quite difficult, costly and dangerous to conduct effective FDO training on an actual surface platform, due to all the uncontrollable variables such as weather, faults and human error. This paper presents a proof of concept air marshalling trainer developed by integrating VBS2 and the Kinect sensor from Microsoft, focusing on the benefits and pitfalls of this approach, as well as lessons learnt.

1. INTRODUCTION

Full motion capture technology has long been a part of high fidelity computer simulation. Motion capture can be used to record human movements so as to develop more realistic virtual avatars, or it can be in real time to determine the position and actions of real participants in the simulation, allowing them to interact more seamlessly with the virtual world.

The Kinect sensor, originally developed by Microsoft as a peripheral for the Xbox gaming console, is an affordable, off the shelf alternative to conventional real-time motion capture and thus presents a unique opportunity for new kinds of interaction with virtual spaces.

2. PURPOSE

The purpose of this research project has been to investigate the efficacy of the Kinect sensor in a military simulation application. In particular, the problem domain of helicopter marshalling was chosen.

The helicopter marshalling domain is a convenient one for a number of reasons.

The set of aircraft marshalling signals is finite, well defined and well understood. The goal is for the trainee to use unambiguous signals so any inflexibility in the resulting system would be seen as a benefit, not a defect

The consequences of mistakes when marshalling are severe, both in human and financial costs. This makes the business case for building a simulation relatively easy to justify.

Helicopters can land on a variety of platforms in the New Zealand context. The cost of training trainees for all these various platforms is high. Fuel, wages and maintenance costs all factor in to the cost of landing a helicopter on (for example) a surface platform while at sea.

Muscle memory is an important part of learning how air marshalling works. Allowing the trainee to interact with their whole body rather than just a mouse makes sense and is not just full body motion tracking for its own sake.

In order to further limit the problem domain, only the following helicopter marshalling signals were investigated: lift off, land, move upward, move downward, move left, move right, move forward, move rearward and hover.

3. METHOD

The method chosen to assess the efficacy of the Kinect was to build and test a prototype. This allows the Kinect and other components of the final solution to be assessed individually and as a whole, resulting in avenues for future research.

For the purpose of this research, the system was designed to land a Kaman SH-2 Seasprite helicopter on the deck of the HMNZS Canterbury. The Canterbury is a multi-role vessel with a helicopter deck, thus enabling it to support helicopter landing and take-off at sea.

4. SYSTEM DESCRIPTION

Figure 1 depicts the components of the prototype system developed for this study; this section describes each component in turn.

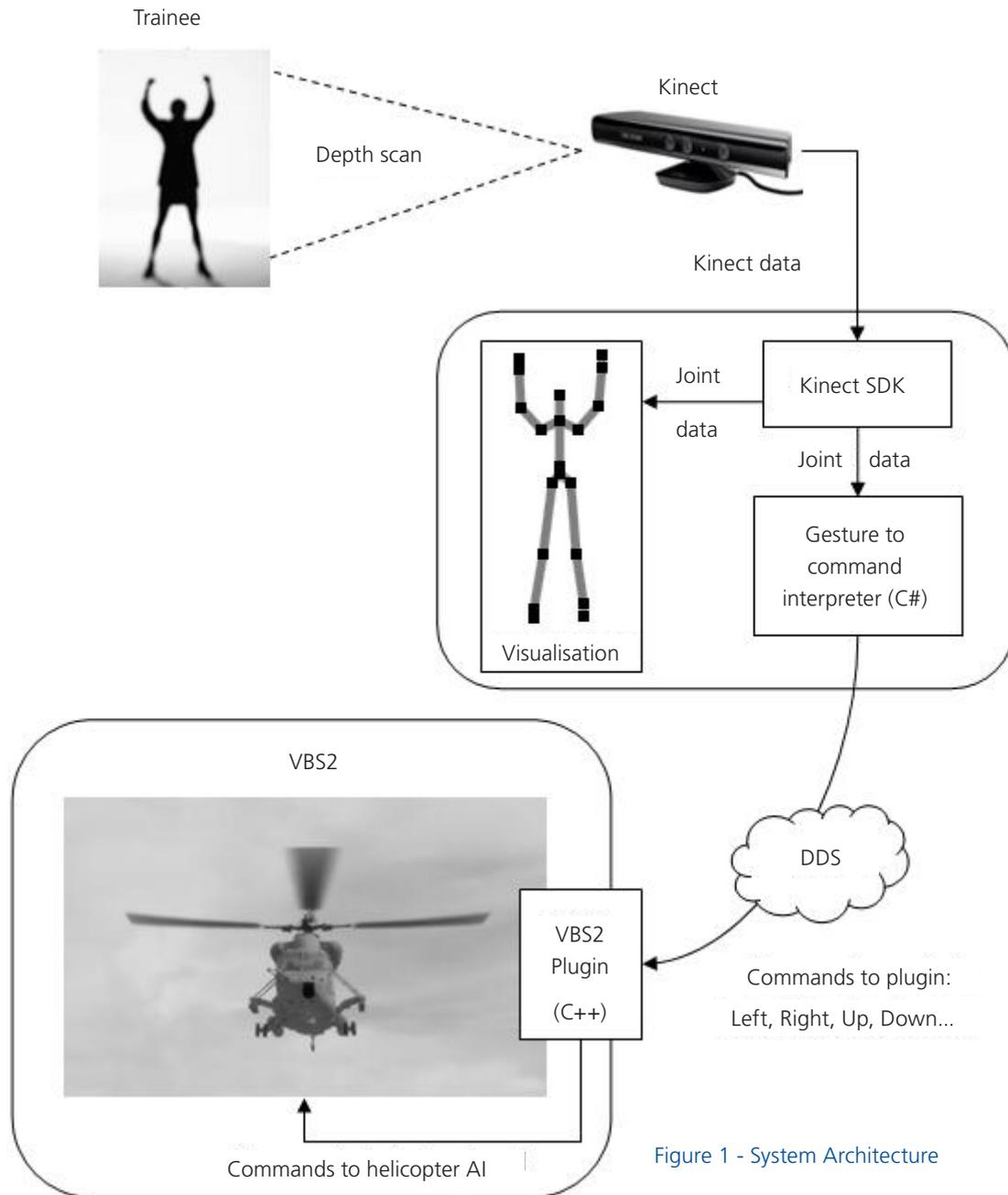


Figure 1 - System Architecture

4.1 Kinect

The Kinect device (formerly Project Natal) was released in North America on November 4th 2010 as a peripheral for the Xbox 360 gaming console.

The Kinect device is a horizontal bar housing a microphone array, an RGB camera, and a depth sensor. For the purposes of this project, the depth sensor is the component of interest.

4.2 Kinect Depth Sensor

The Kinect depth sensor is made up of an infrared laser projector and a monochrome CMOS (complementary metal-oxide-semiconductor) sensor. The infrared laser projector projects an irregular pattern of infrared dots with different light intensities. The CMOS sensor collects the reflected infrared light and reconstructs a depth map by recognising distortions in the known infrared pattern.

The depth sensor can operate at a maximum resolution of 640 x 480 pixels at a rate of 30 frames per second and has a theoretical range of 0.7 to 6 metres. This makes it ideal for real-time full body motion detection.

4.3 Kinect SDK

Microsoft released the Kinect for Windows SDK on Feb 1st 2012. This includes a specially modified Kinect Device which supports USB and a sensor which can operate at both “far” and “near” range, to support users interacting with the device while sitting at a desktop workstation.

The Kinect SDK produces real-time skeleton data, consisting of joints identifying each of the subject’s hands, knees, feet, elbows, and head, positioning them in 3D space with X, Y and Z coordinates. These coordinates can then be mapped to coordinates in the 2D colour video stream also available from the SDK.

The commercial Kinect SDK can track two skeletons simultaneously.

4.4 Gesture Recognition

Gesture recognition involves interpreting movements of the subject’s skeleton as intentional gestures. The Kinect SDK does not ship with gesture recognition and so a suitable algorithm was employed. In this case, a dynamic time warping algorithm was employed.

Dynamic time warping attempts to match two sequences of data. These sequences are “warped” non-linearly in time to determine their level of similarity. The movement of the subject’s skeleton is compared to a number of previous recordings to determine the most similar previously recorded sequence. The most similar sequence is then the “recognised” gesture.

This algorithm was chosen because it is relatively simple to implement and is robust to variability in the speed of joint movements.

4.5 VBS2

VBS2 is a popular, affordable simulation environment, which is a militarised version of the commercially available game Armed Assault by Bohemia Interactive. It contains models for the Seasprite and Canterbury out of the box and supports a plugin interface.

4.6 DDS

DDS (Data Distribution Service) is a real-time publish-subscribe based middleware commonly used to integrate systems across hardware / software boundaries. It is a robust, standards-based technology for integrating across application boundaries, in this case the boundary between the C++ VBS2 plugin and the .Net gesture recognition component.

5. RESULTS

The efficacy of each component is discussed here in turn.

5.1 Kinect

The Kinect performed well, producing a reliable real-time stream of skeleton information in a matter of seconds. There are a few limitations worth mentioning.

5.1.1 Practical Range

The effective range of the device is 1.5 to 3 metres. The device needs to be able to “see” the subject’s feet in order to produce an accurate skeleton. Likewise, the subject’s hands should be visible when held to their furthest extent.

5.1.2 Skeletal joint tracking limitations

The Kinect SDK skeletal tracking algorithm does not appear to perform well when joints cross over each other. For example, when the trainee performs the “land” signal, they cross their arms at their waist. This has the potential to confuse the joint tracking algorithm resulting in poor joint quality. This problem would manifest for any signal that required the trainee to pass their hand across their face or torso as well.

5.1.3 Wrist orientation tracking limitations

The Kinect sensor cannot currently tell the difference between palms faced up and palms faced down. For example, when performing the “move upward” signal, if the trainee drops their arms below shoulder height, this could be confused with the “move downward” signal (with potentially devastating consequences). Since a helicopter pilot would have the same difficulty, this is not an issue for training, but may be necessary for assessment if the trainee is required to hold their hands in a specified manner.

5.2 Gesture Recognition

The Dynamic time warp algorithm was very robust when identifying gestures that involved movement. Up, down, left and right were all reliably detected.

In most cases, gestures involving no movement (more accurately “poses”) were also well detected. Take off, land, and move back were detected well.

5.2.1 Pose versus Gesture

There was, however, an interaction between “hover” and other signals. This is because the move up, down, left and right signals all include the hold signal as a part of their scope of movement. This ambiguity confuses the dynamic algorithm, resulting in false positives, with a bias towards the static “hover” pose since static poses have a lower error.

This identified a need for a multi-algorithm approach. A pose recognition algorithm was developed to identify static poses independently from dynamic gestures.

5.3 VBS2

Movement and appearance of the helicopter was as expected, with a level of performance and fidelity adequate to allow the trainee to interact with the virtual helicopter seamlessly in real time.

Bohemia does not recommend putting anything on the deck of a mobile ship. Doing so causes the ship to list and the helicopter to slide off the deck. Instead a static object representing the Canterbury was used. This has the benefit of being stable and easy to work with but it does not move in the water as a real ship does. This limits the range of scenarios that can be trained for using this approach since “rough seas” will not impact on the stability of the deck.

This limitation is resolved in VBS2 1.6.

6. BENEFITS

6.1 COTS

The solution is commercial-off-the-shelf with some bespoke software development to perform the gesture recognition and for system integration. This reduces cost to develop and maintain.

6.2 Natural gesture recognition

The trainee is not required to wear especially reflective material, coloured balls or special trackers in order to use the system. This makes the system more robust and it lends itself to instructing large volumes of trainees in a short amount of time.

6.3 Muscle Memory

The system enables the trainee to practise with their whole body, facilitating muscle-memory and allowing them to

6.4 Immersion

Engaging the trainee’s whole body in the exercise improves the immersion of the experience.

6.5 Automated Assessment

Using a software-based training platform such as this affords the opportunity to automatically assess the trainee.

7. FUTURE WORK

7.1 Wrist orientation detection

The real-time depth stream could be used to determine the orientation of the subject’s wrists, resulting in a more accurate assessment of their ability.

7.2 Machine learning for tolerances

Machine learning techniques could be employed to derive more fine-grained tolerances for the pose and gesture detection algorithms, resulting in a better “fit” and fewer errors.

7.3 Mobile platform

By upgrading to VBS2 1.6, the surface platform could be made to react more naturally to changes in the environment (such as sea state).

7.4 More realistic helicopter movements

It would improve the training value if the helicopter being marshalled responded to signals in a more natural way. This could be achieved in VBS2 with carefully orchestrated script commands.

7.5 Signal intensity detection

In air marshalling, the speed and force with which the marshalling signals are made indicate how much (or little) a movement is required. The simulation could be extended to adapt the speed of the helicopter’s movement to the intensity of the signal from the trainee.

7.6 More sophisticated signals

The system could be extended to support more complex signals (such as “fire” or “wind direction”).

8. CONCLUSION

Full motion capture is no longer the purview of big budget movie studios. By employing off-the-shelf gaming technology in innovative ways, we can offer trainees a new way to interact with training simulations that more accurately reflects the way in which they learn and work.

REFERENCES

Microsoft Corp. Redmond WA. Kinect for Xbox 360.