RoboMR: A Mixed Reality Robotic Game

Arthur Chomé

Graduation thesis presented for the degree of
Bachelor in Computer Sciences

Faculty of Sciences and Bio-Engineering Sciences
Vrije Universiteit Brussel
Belgium

Contents

1 Abstract

2 Introduction

Video games have been around for a relatively long time. More precisely, it is thought that it all started in October 1958 with the invention of a very simple tennis game by physicist William Higinbotham. It was quite a hit in his research lab and it eventually grew into the game that everybody knows as Pong.

The game industry continued to grow from the 1970's onward. We are now at the 8th console generation with incredibly immersive 3D games like Red Dead Redemption 2 coming out and the PlayStation 4 slowly reaching the end of her life cycle. Meanwhile, a new industry started to emerge: augmented reality.

Augmented reality's forecasted market revenue will be around 80 billion US Dollars by 2021 [2]. Those incredible numbers have drawn the attention of many technology firms that each have by consequence developed their own products for the market. For Microsoft, its newest tool for augmented reality is the HoloLens. This technology now gets used in -for instance- training videos for assembly and manufacturing lines. For us, it will be one of our key technologies for making a mixed reality robotic game. More on that later.

Augmented reality is a relatively new field of research. It involves "augmenting" a person's perception of the real world by adding computer-generated information. This way, a new world is created by "mixing" real-world elements with virtual elements.

This brings us to mixed reality. It's a form of augmented reality that aims to adding virtual objects to the physical world in such a way as they look as if they were really placed within that world. This means there should be a seamless interaction between the physical and virtual elements in the environment. This is what our problem statement will be focusing on.

2.1 Methodology

A methodology is "a system of principles, practices, and procedures applied to a specific branch of knowledge." Such a methodology helps us researchers to produce and present high quality design science research in that is accepted as valuable, rigorous, and publishable in research outlets.

For this research paper, we have chosen to use the Design Science Research Methodology for information systems applied to augmented reality systems[5].

Research activities

This methodology works in a number of clearly defined activities:

1. Problem identification and motivation: To start with, we should clearly define our problem statement: what the problems are and also justify the search for a solution for it. It should be as precise as possible.

2. Define the objectives for a solution: the objectives should be deduced rationally from the problem specification.

3. Design and development: the implementation of the solutions for our designed problem. The final product should be a mixed reality robotic game. This will only be done in the second semester of this academic year.

4. Demonstration: the final developed product should solve the formulated problems. We could apply it on an instance of the problem This could involving the use of a simulation, case study, proof, or other appropriate activity.

5. Evaluation: observe and measure how well the artifact supports a solution to the problem. This activity involves comparing the objectives of a solution to actual observed results from use of the artifact in the demonstration.

6. Communication: this involves informing other researchers about my problem and solutions this paper offers: its utility and novelty, the rigor of its design,etc.

2.2   Problem statement

Since augmented reality is still a relatively small industry with growing stakes, there are some issues in the field we have to account for. We'll formulate our problem statement.

2.2.1   Limited research in the field

Robot navigation based on the vision of head-mounted devices is something new. Therefore, there is still very limited research on the matter. The related work in the following section makes this even more apparent.

2.2.2   Robot vision with limited hardware

To have a mixed reality robotic game, the robot we would use has to be able to see the virtual objects the user can see. The interaction between them should be seamless. The problem here is that robots who have the appropriate sensors on them to track these computer-generated elements have unreasonable price tags. The best we can do is using a robot assembled out of Lego parts.

The robot should be able to see the game objects and interact with them based on what it sees from the head-mounted device of the user. This is for the better since it would crank down the price of the required hardware considerably.

2.2.3   Airborne navigation of virtual objects

The perception problem of the previous paragraph could be applied to the game's virtual objects. This is the prior problem statement but in reverse: instead of worrying about the vision that physical objects have of the virtual objects through the head-mounted device, how about the vision virtual objects have of the physical robot using the HMD vision?

The virtual objects have to hover on top of the targeted robot. For this we should write some logic about aerial navigation but that's not a real problem since writing some basic AI should do the trick. The real issue is properly communicating the robot's position.

2.2.4 Target acquisition

Based on neck movement and eye gaze, how could we optimally steer the robot? We should also be able to communicate some other input like when to shoot, etc.

2.3 Research question

Having dealt with the problems we might bump into, the research question for the paper is as follows: can I make the controlled robot interact with the virtual elements based on the vision it gets from the head-mounted device I am using? In other words: can I navigate the robot with my head-mounted device? If we would find a solution for these questions, it should be as cost-efficient and reliable as possible.

3 Related works

In the previous section, we've talked about our problem statements and reformulated them as research questions. The research papers gathered here focus on the fields of robotics, augmented reality, mixed reality and spatial awareness. This way, we could validate or refine our problem statements. This is very important since making a brand-new paper about something that has already been researched before is redundant. This work should be useful and allow researchers to build upon it.

3.1 Mixed Reality for Robotics[3]

With the ever-increasing presence and growth of robotics in industries and everyday life, it is clear that science has made some real progress in the field. Combining research in robotics with mixed reality has proven to be a great way to come to even more new findings. The central contribution of this work is to establish Mixed Reality as a tool for research in robotics.The 3 specific examples of combining those two disciplines allow us to further elaborate our problem statements.

3.1.1 Benefits

The reason those two unlikely fields got combined is that it brought some very neat advantages for the involved scientists.

One of these advantages is that it eliminates the distance some researchers or research groups may be apart of one another. In other words: no geographical constraints any more. This is thanks to what we call spatial flexibility. Mixed reality allows us to perform experiments with robots remotely. They can meet in a virtual environment instead of having to travel great distances to witness tests on their machines.

Another advantage is the enormous safety benefits linked to using virtual environments. Human-machine interaction would be a thing of the past and interaction would happen solely with virtual models. Potential harm to human test subjects is now impossible.

Mixed reality also makes for an enriched environment where all physical and virtual data interact directly in real-time; no further computation or calculation

is necessary. This expedites and simplifies debugging because the difference between implementation and simulation becomes even smaller.

3.1.2 Experiment 1

Using camera's mounted on unmanned aerial vehicles the research team conducted three kind of experiments with robots. It is especially the third one that is interesting here. The first experiment focused on making drones follow a human around. But since human motion is complex and unpredictable, and UAVs are highly dynamic systems, the typical coarse-detail simulation is not expected to be very accurate.These issues may pose safety risks for humans sharing the environment with the robots. This is why they used mixed reality to simulate the movement of a human using Unity.

3.1.3 Experiment 2

The next experiment involved spreading a limited amount of drones on a wide area that constantly changed using mixed reality. This is interesting since we can see drones adapt on the fly on changing mixed environments. It allows for swarm algorithms to be tested in a simulation that more closely depicts the actual physical motions of a robot, prior to implementing the algorithms on their real counterparts.

3.1.4 Experiment 3

Finally, a camera-equipped UAV can be used to identify and track robots using special markers (so-called augmented reality tags) and vision processing. The function of the robots is to move a box to a specific location. Based on the locations of the robots and the box, which are computed with vision processing from the drone's camera view, the robots drive toward the box to move it. While this is a simple scenario, it demonstrates the capabilities of the mixed reality approach for testing vision-based approaches without having the required set of hardware.

3.1.5 Problem statement review

While the 3 experiments are interesting applications of mixed reality, the most interesting one would be experiment 3: the drone actually moves other robots by using its perception of the world. The camera on the drone can be easily generalized to a head-mounted device like the Microsoft HoloLens that would also use a similar technology like AR tags and image processing. The only challenge left would be to make the robots interact with virtual objects instead of a physical box.

3.2 Pinpointing: Head- and Eye-Based Target Selection[4]

A next important aspect in our mixed reality robotic game is the concept of target acquisition neck movement and eye gaze.Head movements are deliberate and accurate, and provide the current state-of-the-art pointing technique. Eye gaze can potentially be faster and more ergonomic, but suffers from low accuracy due to calibration errors and drift of wearable eye-tracking sensors. This work

5

investigates precise, multimodal selection techniques using head motion and eye gaze.

3.2.1 Problem statement review

Target acquisition is a big deal for the theses. Without it, the game wouldn't even exist since there's no way to shoot down the enemy aircraft. The paper displayed a lot of techniques with eye gaze and neck movement to select targets. It actually offers all the answers to our problem statement about target acquisition.

3.3 Accurate real-time occlusion for mixed reality[6]
3.4 A real-time tracker for markerless augmented reality[1]

This paper addresses the problem of implementing markerless realtime augmented reality. A form of AR where the used application doesn't need any prior knowledge of the physical environment to overlay 3D content into a scene and hold it to a fixed point in space.

This is quite a novelty in augmented reality and is attributed to the recent emergence of better cameras and other sensors. These sensors were perfected so much so that they were soon able to -for instance- detect locations of walls and points of intersection without needing any foreknowledge.

Like mentioned before, markerless AR can be applied on a variety of sensors, be it visual or not. This paper focuses on a monocular vision sensor or a camera to keep it simple. This is in our best interest since the MicroSoft HoloLens is a camera mounted on a user's head. This study will focus on the registration techniques that allow alignment of real and virtual worlds using images acquired in real-time by a moving camera. The paper compromises 4 experiments where

3.4.1 Virtual Visual Servoing

The contribution of this paper is the design of a virtual visual servoing system. It strongly resembles normal servoing: a technique where a robot gets controlled and steered by using feedback information extracted from a visual sensor like a camera. But in this case, we will not be correcting a robot but virtual objects based on their current positions in the real world as seen by our head-mounted device.

3.4.2 Experiment 1: Tracking in an indoor environment

The first experiment involves tracking in an indoor environment. With tracking, we mean that the used visual sensor or camera keeps an eye on elements in the physical environment. Specific to this experiment, it has to track four 3D circles. Although the images are quite simple in this experiment, if no robust estimation is considered tracking fails after a few images because the minimization process has to deal with miss-tracking and problems due to occlusion.

3.4.3 Experiment 2: Tracking in an outdoor environment

A little bit trickier than previous experiment since outdoor environments provide more environment issues. The experiment had to track a cilinder and correctly place a virtual object relative to it. Despite very noisy images (wind in the trees, multiple occlusions, etc.) tracking was still achieved.

3.4.4 Experiment 3: An AR application for system maintenance

The experiment here recreates a guided maintenance system for an air-conditioning system. The scenario deals with heavy occlusions and the effects of video inter-lacing.scenario. It turns out that the realtime AR algorithm is very robust to many sources of external error. So much so that in spite of heavy occlusion and disturbances, tracking is still very reliable and handled in realtime.

3.4.5 Experiment 4: Tracking in a 3D visual servoing experiment

A positioning task using a couple-charged device camera has been also considered. A CCD camera is a type of image capture device that utilizez an image sensor to register visible light as an electronic signal. The visual sensor is mounted on a robot that would start in an initial position and make his way to a desired position expressed as a desired position of the object in the image. The result of the tracking algorithm was very positive.

3.4.6 Problem statement review

This article was chosen to tackle the problem statement of airborne navigation for the game's virtual objects. Specifically, the theses requirement states the objects should hover on top of the robot. Using virtual visual surviving and some basic AI algorithms this should be feasible.

The article confirms that we can correct the position of the game objects by way of a moving camera perspective, in this case the Microsoft HoloLens. The head-mounted device would be using markerless augmented reality to keep track of the physical robot and the technique of virtual visual servoing would correctly place the virtual objects relative to it.

The only question remains how to combine these technologies with the navigation logic of the hovering elements. But this is more of an implementation issue.

4 Mixed Reality Robotic Game

4.1 Overview

The purpose of the final bachelor theses is to realize a mixed reality robotic game. This means a new environment will be created by merging virtual elements to the physical world we can observe. A physical robot will be controlled by a user using a head-mounted device. Some virtual objects will be hovering over the robot and the user has to steer the robot using eye gaze to shoot them down. To add to the game play experience, some must-have game elements like high-scores, player upgrades or a difficulty setting will be added.

4.1.1 Head-Mounted Device

With the help of immersive technologies like the Microsoft HoloLens we will be able to display our virtual elements in the environment for the user to see. The user of our application will interact with the virtual elements by using a physical, remotely controlled robot built out of Lego blocks.

4.1.2 Target Acquisition

With the help of eye gaze and neck movement, the user should be able to select a virtual object -an enemy- and instruct the robot to direct its firing on it.

4.1.3 Airborne navigation

Airborne navigation is a big part of the thesis: game objects will be hovering on top of their targets so some basic artificial intelligence will have to be programmed.

4.2 Implementation

The implementation of the game will be handled in the second semester of this academic year. Though not an urgent matter to consider, it's still handy to think about the technologies that will be used.

4.2.1 Programming language: C Sharp

C Sharp is a high-level programming language introduced by Microsoft and built as an extension of the C programming language (just like C++). It is used in essentially all their products including the Microsoft HoloLens. Easy to say that -to write an app for this device- using C Sharp is a prerequisite we have to meet.

4.2.2 Unity as 3D Engine

Unity is a cross-platform game engine that can be used to implement both 2D and 3D games. It's widely used by amateur game enthusiasts and boosts an excellent IDE that makes the whole coding experience easy and enjoying. We will use its Integrated Development Environment to write our C Sharp code.

4.2.3 Microsoft HoloLens

The Microsoft HoloLens is one of the first commercially available head-mounted devices on the market. It's quite expensive costing around 3000 US Dollars and about 5000 for the commercial suite. There might be some difficulties using it keeping in mind it's quite a novelty on the market.

4.2.4 Mixed reality platform

As part of the Windows 10 operating system, Microsoft released with it Windows Mixed Reality. This is a platform that provides holographic and mixed reality experiences with compatible head-mounted displays like its flagship product the Microsoft HoloLens. It will be an invaluable asset for the theses since it eases the integration of our software with the HoloLens.

5 Conclusion

5.1 Conclusion

5.2 Acknowledgements

References

Comport, A. I., Marchand, É., & Chaumette, F. (2003). A real-time tracker for markerless augmented reality. , 36.

Evans, G., Miller, J., Pena, M. I., MacAllister, A., & Winer, E. (2017). Evaluating the microsoft hololens through an augmented reality assembly application. , *10197*, 101970V.

Hoenig, W., Milanes, C., Scaria, L., Phan, T., Bolas, M., & Ayanian, N. (2015). Mixed reality for robotics. , 5382–5387.

Kytö, M., Ens, B., Piumsomboon, T., Lee, G. A., & Billinghurst, M. (2018). Pinpointing: Precise head-and eye-based target selection for augmented reality. , 81.

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of management information systems*, *24*(3), 45–77.

Walton, D. R., & Steed, A. (2017). Accurate real-time occlusion for mixed reality. , 11.