Evaluating the Effectiveness of Mixed Reality Simulations for Developing UAV Systems

Ian Yen-Hung Chen[1], Bruce MacDonald[1], and Burkhard Wünsche[2]

[1] Dept. of Electrical and Computer Engineering,
University of Auckland, New Zealand
{i.chen, b.macdonald}@auckland.ac.nz
[2] Dept. of Computer Science,
University of Auckland, New Zealand
burkhard@cs.auckland.ac.nz

Abstract. The development cycle of an Unmanned Aerial Vehicle (UAV) system can be long and challenging. Mixed Reality (MR) simulations can reduce cost, duration and risk of the development process by enabling the replacement of expensive, dangerous, or not yet fully developed components with virtual counterparts. However, there has been little validation of such hybrid simulation methods in practical robot applications. This paper evaluates the use of MR simulations for prototyping a UAV system to be deployed for a dairy farming monitoring task. We show that by augmenting the robot's sensing with a virtual moving cow using an extensible Augmented Reality (AR) tracking technique, MR simulations could help to provide efficient testing and identify improvements to the UAV controller. User study findings reveal the importance of both virtual and MR simulations to robot development, with MR simulations helping developers transition to development in a more physical environment.

1 Introduction

Unmanned Aerial Vehicles (UAVs) are ideal for livestock and vegetation monitoring in agriculture. The high mobility of UAVs enables fast exploration of agriculture fields to collect information in remote areas that are otherwise difficult to access by farmers. The development of UAV systems is however difficult, especially during testing due to potentially dangerous operations, site availability, weather conditions, and considerable resource requirements.

Mixed Reality (MR) simulation [4] can provide a cost-effective solution to robot experimentation. The simulator is founded on the concept of MR [11] and enables developers to design various test scenarios for evaluating robot systems involving real and virtual components. There have been increasing interests in applying MR to robotics. The literature reveals that similar work exists in using hybrid or MR simulation techniques for developing robot systems in areas such as humanoid robotics [13, 12], underwater robotics [7], and aerial robotics [8]. Existing work proposes that the use of these simulation techniques benefits development in terms of cost and safety, however, efforts are primarily limited to conceptual designs and software implementations with minimal evaluations.

This paper presents a case study evaluation of MR simulations for prototyping a vision-based UAV system to be deployed for a cow monitoring task in agriculture. There are novel challenges in creating MR simulations for this task. We assess our MR robot simulator's [4] capability to a) provide robot developers a safe and efficient way of creating test scenarios for this high-risk operation, and b) accurately augment the robot's visual sensing with virtual inputs under the erratic motion of the UAV platform. A user study is also conducted to examine the robot developers' use of MR simulations for prototyping UAVs, and an analysis of its results forms one of the main contributions of this paper. Similar user studies have been presented such as in [6], but the focus is on the visualisation aspects of MR for robot development. Our previous work [5] investigated the user's perception of the MR simulation technology and its visual interfaces in a robot operation task, which did not involve any implementations. In comparison, this paper describes a user study that identifies how MR simulations can help robot programmers in implementing a robot software component, the development stages and tasks which MR simulations are suitable for, and how they compare with virtual simulations.

The remainder of the paper is organised as follows. Section 2 gives background on the cow monitoring project. Section 3 describes the MR simulation created for testing the UAV prototype. Section 4 presents evaluation results. Section 5 describes the user study and its findings.

2 Cow Monitoring Project Background

One of the main hurdles of efficient dairy farming is the lack of support for monitoring and tracking cow status on the farm. The farmers are interested in information such as a cow's health status and temperature (for indicating signs of pregnancy or oestrus), which affect its productivity. However, a single farmer is typically responsible for looking after a herd size of several hundreds of cows, which makes the process of information management difficult. In response, a vision-based cow monitoring UAV system is proposed for this task. As a proof of concept, the development began with a micro aerial vehicle. The Hummingbird quadrotor from Ascending Technologies [1] is used. Hummingbird comes with an onboard controller that implements attitude stabilisation based on the onboard IMU. A Point Grey Firefly MV camera with 4mm lens is mounted directly beneath the centre of the quadrotor and oriented to look downwards for the cow monitoring task. 640x480 colour images are transferred over the firewire cable to the ground station where the computations are performed. The goal is to enable the UAV to autonomously detect a target cow on the ground using the onboard camera and hover over the target so that any data of interest can be collected by specialised instruments carried onboard the UAV. The quadrotor should try to stay hovering above the target and follow its movement for the data collection process. The development is at the prototyping stage, and testing is primarily carried out in an indoor environment. Figure 1 shows the robot in the experimentation space.

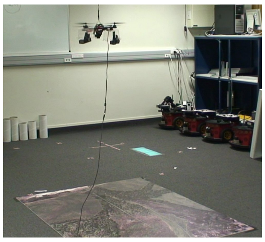


Fig. 1. Indoor UAV system test setup on a mock-up agricultural environment. The experimental setup is not to scale.

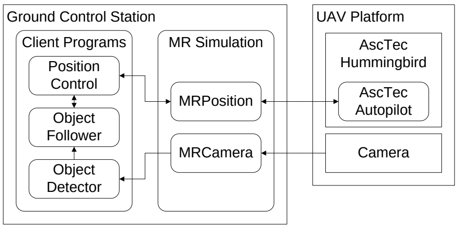


Fig. 2. MR simulation system diagram for testing the UAV prototype.

Three prototype components have been implemented: 1) The **Object Detector** processes image data and identifies target location in image coordinates (x_i, y_i) using a colour blob detection algorithm. 2) The **Object Follower** generates position commands in world coordinates (x_w, y_w, z_w, ψ_w) based on the target's image location to keep it in the view of the robot. The **Position Control** implements Proportional–Integral–Derivative (PID) controllers for robot's x, y, z, and yaw motion. Controller feedback is in the form of vision pose estimates, notably using the Augmented Reality (AR) tracking algorithm described in Section 3.1 to provide the pose estimates in a similar way to [2].

3 Mixed Reality Simulation

The project was in need of an efficient and cost-effective method of testing the prototype UAV system. Conducting field tests in this early stage is considered infeasible as it could harm the real animal and the surrounding agricultural objects in case of failures. MR simulation is able to help in testing the UAV system by simulating a virtual animated model of the target cow to be followed. It provides developers complete control over the virtual target and enables them to efficiently create repeatable tests of various scenarios, e.g. moving pattern and speed of the cow. To provide as much insight to real world tests as possible, the real UAV is used. The MR simulation enables the robot developers to observe the actual UAV flight behaviour as it responds to the simulated input in real time. This is believed to have a benefit over testing using pre-recorded videos.

3.1 Augmenting the Robot's Sensing

To facilitate interaction between the real robot and the virtual cow, the primary task is to augment the robot's vision input to reflect the presence of this virtual cow. This task is, however, difficult due to the erratic motion of the UAV platform. Registration of the virtual cow must be accurate since the behaviour of the robot is directly dependent on the resulting augmented image data produced by the MR simulation. Large errors in augmentations could lead to unexpected UAV movements and cause a serious crash.

To achieve this real-virtual interaction, we integrate an extensible AR tracking solution based on the Parallel Tracking and Mapping (PTAM) algorithm [9]. PTAM takes a structure-from-motion approach to track the camera while building a map of the environment in real time. Tracking is performed efficiently in a separate thread from mapping, relying on a large number of FAST corners and a motion model. The mapping thread incrementally builds the map from carefully selected image frames, known as keyframes, to ensure quality, and applies bundle adjustments to refine the pose estimates of the map elements and keyframes.

However, PTAM's manual map initialisation process creates the base map in an inconsistent scale and an unpredictable origin. We modify the original map initialisation process to associate the two initial keyframes (which triangulate the base map) with pose estimates from the markerless AR algorithm described in [3], allowing the map to be constructed in the real world metric space and coordinate system defined by the user. Pose estimates from the extensible AR system are used for registering the virtual cow onto the live video imagery. This generates augmented image data that will be delivered to the Object Detector component of the UAV system.

A system diagram of the components in this MR simulation is shown in Figure 2. Instead of reading images from the real camera, the Object Detector processes augmented images generated by MRCamera for cow detection.

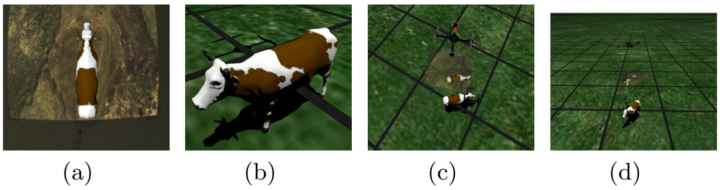
3.2 MR Interfaces

MR interfaces are provided for monitoring the MR simulation, as shown in Figure 3. The AR interface provides users a view of the environment from the robot's onboard camera, reflecting the presence of the virtual cow. On the other hand, an Augmented Virtuality (AV) interface is designed to offer multiple views of the simulation from virtual perspectives.

4 Evaluation

The MR simulation was built on the MR robot simulator detailed in [4]. To assess its performance in simulating the prototype UAV system, an experiment was conducted that involved the robot following a moving virtual cow. The purpose is to investigate whether the augmentation in the robot's vision sensor data would be sufficiently robust and accurate to facilitate reliable robot responses. The virtual cow was animated to move in three different maneuver patterns within the space covered by the aerial map on the floor. Pose estimates of the robot given by the extensible AR tracker were recorded as it followed the cow.

The simulation results are shown in Figure 4. The robot followed the cow in each case, demonstrating successful interactions between the real robot and the virtual cow. The extensible AR tracker was able to cope with several sudden



(a) (b) (c) (d)

Fig. 3. a) The AR interface showing a view of the scene from the robot's perspective. b) The AV interface with free-look camera mode: in this example, the virtual camera is moved to focus on the target cow to be monitored. c) The AV interface with tethered camera mode: the virtual camera follows and tracks the movement of the UAV. d) The AV interface with fixed camera mode: similar to the tethered camera mode but with a fixed virtual camera.

and erratic motions of the UAV and accurately augmented the virtual cow in the view of the camera. The AR tracking errors were measured to be approximately 0.02m in x, 0.03m in y, and 0.06m in z. Figure 5 shows screenshots from the Object Detector window, which displays the processing being carried out on the augmented vision data generated by the MR robot simulator. The virtual cow remained aligned with the real world scene as it was animated to move across the aerial map. While no significant errors were observed in the pose estimates, it was noticed that the extensible AR tracker picked up fewer points in scenes with large portions of grasslands due to the lack of textured content. This is an expected behaviour of the underlying PTAM algorithm used. Nevertheless, there were sufficient textured patches surrounding this grassland region in this experiment to maintain the tracking quality. It must be noted that future outdoor experiments involving scenes with very few distinct features may cause the algorithm to produce poor results.

Overall, the use of the MR simulation helped to provide initial insights into the behaviour of the prototype UAV system for this cow monitoring task. The UAV trajectories in Figure 4 (especially the sine wave pattern) revealed to the robot developers the flight instability of this prototype UAV system due to the poor control logic of the object following algorithm, identifying the need for improvement.

5 User Study

Our user study examines the user's development approach and preferences in using MR simulations for implementing robot software components. It targeted a user group of experienced robot developers and computer programmers who were required to write software code for controlling the UAV and test their program in simulations. The participants were provided with three development methods, including MR simulations, and it was up to the participants how they used them. We acknowledge that the task given may take the participants through different

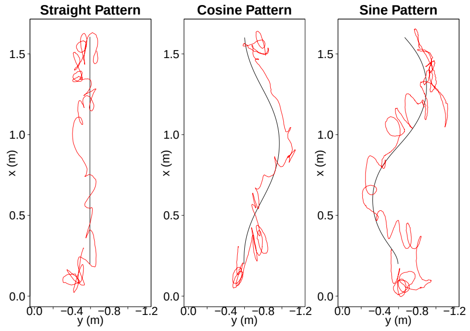


Fig. 4. UAV flight paths (red) from following three moving patterns of the virtual cow (black).

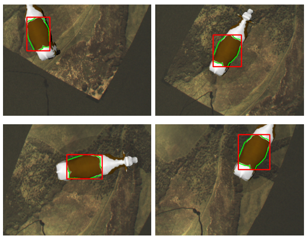


Fig. 5. Screenshots illustrating the image processing carried out by the Object Detector. Simple colour detection was implemented as a proof of concept for cow detection.

development stages, and the experimental setup may need to vary accordingly to meet their needs. Instead of restricting the participants to a single method for the whole development process, it is more practical to let them experiment freely using the provided setups. The participants were video-taped for post-study analysis. Desktop videos were also taken to examine their actions in using the MR simulation tool and its interfaces.

5.1 Task

The participants occupied the role of a robot programmer. The primary task was to implement an algorithm for controlling the robot's movement and test their implementation in simulation. They were given the fully implemented Object Detector and Position Control components, and a skeleton code of the Object Follower component which they had to complete. The algorithm had to generate position commands for moving the robot in order to keep the detected target (given in image coordinates) in its view. Three development methods were provided, two of which are MR simulations.

1. **Method One (Virtual simulation):** Simulation is provided by Gazebo [10]. No complex aerodynamics is integrated in the simulation. A simple dynamics model is used to mimic the behaviour of the UAV under the control of the PID controller.
2. **Method Two (Free-moving Camera):** This setup involves manual movement of the onboard camera over the aerial map. A virtual cow is placed in the environment and augmented images are transferred to the UAV system. The AV interface reflects the UAV movement in real time.
3. **Method Three (UAV + Camera):** This setup involves the real UAV taking flight over the aerial map under the control of the Position Control component. A virtual cow is placed in the environment and augmented im-

ages are transferred to the UAV system. The AV interface reflects the UAV movement in real time.

The participants were free to switch between the development methods for testing their implementation as the development progressed. Sample test programs for moving the virtual cow were provided. They were asked to keep in mind the stage of development which they considered themselves at when using the simulations, assuming the development cycle could be divided into:

1. **Stage One:** Initial implementation and logic validation
2. **Stage Two:** Debugging and fixing errors in code
3. **Stage Three:** Evaluating performance of algorithm
4. **Stage Four:** Tuning and refining parameters

5.2 Procedure

The cow monitoring project was first explained to the participants, and the design of the UAV system was shown. Before starting the task, a quick demonstration of the three simulation methods was given. When a participant indicated task completion, the working of the system was verified in a test run that tested whether the UAV could correctly follow the virtual cow moving in a sine wave maneuver pattern. If the UAV failed to follow, the participant was given a chance to refine their code and conduct further testing. A second and final test run was held upon indication by the participant. A maximum of two and a half hours were given for the task.

5.3 Questionnaire

The questionnaire comprised four sections. Section one collected the participant's demographic information. Section two collected the participant's time distribution in using the provided development methods over the four stages of the development cycle. Section three measured the participant's experience in using the different development methods on a 7-point Likert scale. The last section collected the participant's preferences in using the development methods.

5.4 Hypotheses

1. The participants would transition to a more physical experimental setup as the development progresses. Method One and Method Three are believed to be essential, while Method Two benefits users who take a more cautious and systematic approach to testing.
2. Method One is useful for initial implementation (Stage One) and debugging errors (Stage Two). It may also be useful for experimenting with different object following strategies (Stage Three).
3. Method One is safer and easier to use compared to other methods. However, it provides the participants lower confidence in the actual behaviour of the UAV system in comparison to Method Three.

4. Method Two is useful for helping users understand the robot's field of view, and for validating the position commands (Stage One and Stage Two) by comparing them against the real world experimentation area before commencing test flights.
5. Method Two may also be used as a safer alternative for tuning and refining the control parameters of the algorithm (Stage Four).
6. Method Three is heavily used in the later development stage for evaluating performance (Stage Three) and tuning and refining parameters (Stage Four).

7. Method Three is effective and results in fewer mistakes. The risk of test flights is also higher and the method may require a longer learning curve.

5.5 User Study Results

10 participants were recruited (4 academic researchers, 5 postgraduate students, 1 software engineer), all of whom were experienced computer programmers (mean 8.95 years of experience, SD 6.82). 7 participants had experience in robot development (mean 3.93 years of experience, SD 2.05). None had experience in developing aerial robot systems. The 10 participants are coded P1, P2, P3, etc.

5.6 Participant's Development Approach

9 participants successfully completed the task within the given time. The remaining participant, P3, reported having struggled to understand the relationship between the different coordinate systems, a respect in which the development methods and the MR interfaces were unable to help the user.

More than half of the participants (6 participants) did not choose to use Method Two. Responses were because the participants believed a) the method was not useful for developing control algorithms (4 comments), and b) the same benefit of this method could be obtained using Method One and/or Method Three (2 comments). Interestingly, different opinions were collected from 2 participants who used Method Two. P1 and P8 commented that it was necessary to see the real environment to understand the robot's field of view when determining the starting control parameters. The non-responsiveness of the robot in this method was considered beneficial and they were able to debug their algorithms while the robot stayed in a particular position relative to the target. The findings supported hypothesis 4 to only some extent since only a minor proportion of the participants used this method. A useful comment was that the benefit of using Method Two could be leveraged if the simulation provided readily available graphical aids to help visualise the position commands in the MR interfaces.

Observations found that 9 participants began with Method One. P8 was the only exception and decided to use Method Two to collect all the parameters and implement the algorithm before testing it using Method One. All participants used Method Three as the last development method. The results suggest that the participants slowly transitioned from virtual simulation to MR simulation (with the real UAV) as the development progressed, supporting hypothesis 1.

Stage \ Method	**M1**	**M2**	**M3**	Subtotal
Stage One	**16.5%**	1.0%	3.5%	21.0%
Stage Two	**7.8%**	0.6%	6.1%	14.5%
Stage Three	4.5%	3.0%	**14.5%**	22.0%
Stage Four	6.5%	1.5%	**34.5%**	**42.5%**
Total				100%

Table 1. Time distribution in using the three development methods (Method One: M1, Method Two: M2, Method Three: M3)

Stage \ Method	**M1**	**M2**	**M3**
Stage One	**10**	1	0
Stage Two	**9**	2	2
Stage Three	6	2	**9**
Stage Four	2	2	**9**

Table 2. Users' recommended development methods for approaching each stage of the development cycle.

The average percentage of time spent in each development method at each stage is shown in Table 1. Participants relied heavily on using Method Three for tuning and refining control parameters of the algorithm (34.5%). Observations also found that using virtual simulation is still important in the evaluation stage, with a majority of the participants (8 participants) constantly switching back to virtual simulation for evaluating their object following strategies. P8 pointed out that the time for using Method Three was precious since it required more time to set up and also consumed battery resources, and therefore, Method One was a cheaper and faster alternative for testing. Table 2 shows the user preferences if they were to develop a similar robot program in the future. Their recommendations conform to the time distribution findings. Method One was the most preferred method for initial implementation and debugging, while Method Three was more suitable for evaluating performance and tuning parameters. The results supported hypothesis 6 and the first part of hypothesis 2.

5.7 Monitoring UAV Operations

Reviewing the recorded desktop videos identified that all participants relied on the first-person AR view for evaluating the robot's performance, either by focusing on the AR interface or the Object Detector's window which displayed the same view. This enabled them to evaluate whether the robot was able to achieve the goal of the task, i.e., keeping the target object in the view, while ensuring the aerial map remained in the view for the Position Control component to function correctly. It was observed that participants who used the AV interface (4 participants) mainly chose the free-look camera and/or the tethered camera mode. However, 3 participants commented that although the interfaces helped to spot unusual UAV behaviour, reacting in time to prevent a crash was difficult and required experience. Again, this suggested the need for additional visualisations to help users foresee and take actions to prevent failures.

5.8 Statistical Analysis

The questionnaire results on the development method experience are shown in Figure 6. Method One and Method Three were rated by all the participants, and

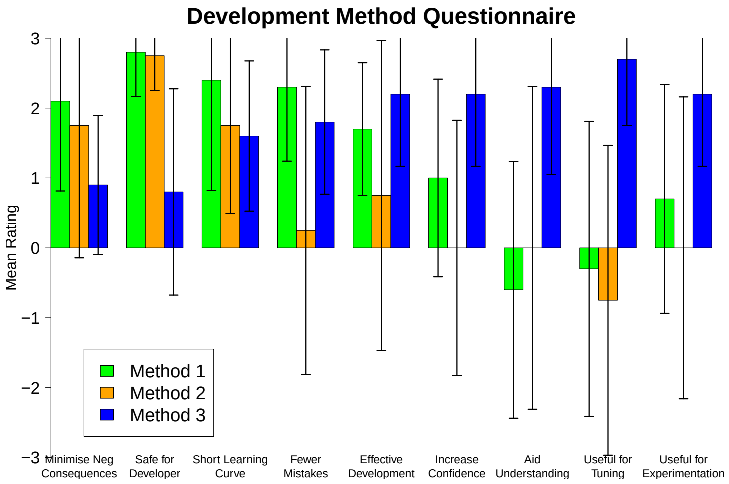


Fig. 6. Users' experiences in using the three development methods.

Method Two was rated by the 4 participants who used this method. As Method Two had very different characteristics from the other two conditions, statistical tests were only performed to identify significant differences between Method One and Method Three. Method One and Method Three data were compared using Two-tailed Wilcoxon Signed Rank tests, while user ratings on Method Two provided an indication of how the method fits the related hypotheses.

As expected (hypothesis 3), using a real UAV had a greater impact on safety. The participants rated Method One significantly higher than Method Three for keeping negative consequences of failures minimal (Z = -2.20, $p < 0.05$), and being safe for the developer (Z = -2.87, $p < 0.05$). Nevertheless, the average ratings for Method Three in terms of safety are overall positive.

The results did not indicate that Method One had a significantly shorter learning curve than Method Three (Z = -1.54, $p = 0.12$), although it should be noted that Method Three required more preparation steps, such as initialising AR and starting an extra Position Control process, which took more effort for the users to set up the simulation.

The results did not indicate that Method Three was significantly more effective than Method One (Z = -1.00, $p = 0.32$). Similarly, there was no significant difference between Method Three and Method One for giving users greater confidence (Z = -1.56, $p = 0.12$), or making fewer mistakes in the development process (Z = -0.91, $p = 0.36$). Consequently, the results did not support hypothesis 7. However, observations and post-study video analysis identified that an unexpected hardware influence on the UAV behaviour in Method Three may have led some participants to create mistakes during development. Upon transitioning to Method Three, the majority of the participants found their algorithm did not produce reliable robot performance and began to make incorrect changes to their code before performing additional testing. The true cause for this UAV behaviour was due to the effect the camera cable had on the x-movement of the UAV, which was not simulated in Method One. Unfortunately, MR simulations and the interfaces were not designed to help them isolate this problem.

The participants rated Method Three significantly higher than Method One for understanding the UAV flight behaviour as expected (Z = -2.57, $p < 0.05$), since it involved the real UAV platform in simulation. The finding could change if a high-fidelity flight dynamics model was integrated into the virtual simulation. This needs to be investigated in the future.

Method Three was rated significantly higher than Method One for tuning control parameters (Z = -2.81, $p < 0.05$). The findings agreed with the development method time distribution data analysis and the user recommendations, suggesting Method Three was more suitable for the later stages of the development; this further supported hypothesis 6.

Statistical analysis did not suggest that Method Three was more useful than Method One for experimenting with different flight maneuvers (Z = -1.80, $p = 0.07$), though the result was approaching significance. While both average ratings were positive, there was high variance in Method One ratings, thus the results could not fully support the second part of hypothesis 2. Method Two was found to be unsuitable for tuning, which rejected hypothesis 5. P1 and P8 commented later in the interview that Method Two was useful for *identifying* (not tuning) initial control parameters, which were later tuned using Method Three.

5.9 Discussions

The results on the overall participant's development approach may not seem surprising but there were specific findings which were not expected. In this study, statistical analysis did not indicate that the use of MR simulation (with real UAV) was significantly more effective than virtual simulation or was it more useful for experimentation. The findings were however not unfavourable because MR simulation does not intend to replace existing simulation methods but to complement them during the robot development process; a hypothesis that was supported by the results. Further research is necessary to validate the strengths of MR simulations in a more focused, long term study that compares the use of MR simulation alone against using only virtual simulation.

6 Conclusions

This paper contributes a case study evaluation on using MR simulations for the development of a prototype UAV system. The simulation augmented the robot's onboard vision data with virtual inputs using a modified PTAM algorithm for testing the cow monitoring operation. The use of the MR simulation has been shown to help provide valuable insights into the UAV system performance that are otherwise difficult to obtain in real world tests. User study results indicated that MR simulations complemented virtual simulations, providing an intermediate experimental environment before moving onto testing in the real world. It helped robot programmers understand UAV flight behaviours and tune control parameters for better robot performance. Future work includes providing more visualisation aids to improve monitoring of UAV tasks, and conducting a user study to identify cases where the benefits of using MR simulations are more evident compared to traditional simulation approaches.

References

1. Ascending Technologies, http://www.asctec.de/
2. Blösch, M., Weiss, S., Scaramuzza, D., Siegwart, R.: Vision based MAV navigation in unknown and unstructured environments. In: Proceedings of the IEEE International Conference on Robotics and Automation. pp. 21–28 (2010)
3. Chen, I.Y.H., MacDonald, B., Wünsche, B.: Markerless augmented reality for robots in unprepared environments. In: Proceedings of the Australasian Conference on Robotics and Automation. Canberra, Australia (December 3–5 2008)
4. Chen, I.Y.H., MacDonald, B., Wünsche, B.: Mixed reality simulation for mobile robots. In: Proceedings of the IEEE International Conference on Robotics and Automation. pp. 232–237. Kobe, Japan (May 12–17 2009)
5. Chen, I.Y.H., MacDonald, B., Wünsche, B., Biggs, G., Kotoku, T.: Analysing mixed reality simulation for industrial applications: A case study in the development of a robotic screw remover system. In: Proceedings of the Second International Conference on Simulation, Modeling and Programming for Autonomous Robots. pp. 350–361. Darmstadt, Germany (November 15–18 2010)
6. Collett, T., MacDonald, B.: An augmented reality debugging system for mobile robot software engineers. Journal of Software Engineering for Robotics 1(1), 18–32 (2009)
7. Davis, B., Patron, P., Lane, D.: An augmented reality architecture for the creation of hardware-in-the-loop & hybrid simulation test scenarios for unmanned underwater vehicles. In: OCEANS. pp. 1–6 (2007)
8. Göktoğan, A., Sukkarieh, S.: An Augmented Reality System for Multi-UAV Missions. In: Proceedings of the Simulation Conference and Exhibition, SimTect. Citeseer, Sydney, Australia (May 9–12 2005)
9. Klein, G., Murray, D.: Parallel tracking and mapping for small AR workspaces. In: Proceedings of the Sixth IEEE and ACM International Symposium on Mixed and Augmented Reality. pp. 225–234. Nara, Japan (November 2007)
10. Koenig, N., Howard, A.: Design and use paradigms for Gazebo, an open-source multi-robot simulator. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems. vol. 3, pp. 2149–2154 (September 28–October 2 2004)
11. Milgram, P., Colquhoun, H.: A taxonomy of real and virtual world display integration. Mixed Reality-Merging Real and Virtual Worlds pp. 5–28 (1999)
12. Nishiwaki, K., Kobayashi, K., Uchiyama, S., Yamamoto, H., Kagami, S.: Mixed reality environment for autonomous robot development. In: Proceedings of the IEEE International Conference on Robotics and Automation. pp. 2211–2212. Pasadena, CA, USA (May 2008)
13. Stilman, M., Michel, P., Chestnutt, J., Nishiwaki, K., Kagami, S., Kuffner, J.: Augmented reality for robot development and experimentation. Tech. Rep. CMU-RI-TR-05-55, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA (November 2005)