

The Effect of Registration Error on Tracking Distant Augmented Objects

Mark A. Livingston*

Zhuming Ai†

Naval Research Laboratory

ABSTRACT

We conducted a user study of the effect of registration error on performance of tracking distant objects in augmented reality. Categorizing error by types that are often used as specifications, we hoped to derive some insight into the ability of users to tolerate noise, latency, and orientation error. We used measurements from actual systems to derive the parameter settings. We expected all three errors to influence users' ability to perform the task correctly and the precision with which they performed the task. We found that high latency had a negative impact on both performance and response time. While noise consistently interacted with the other variables, and orientation error increased user error, the differences between "high" and "low" amounts were smaller than we expected. Results of users' subjective rankings of these three categories of error were surprisingly mixed. Users believed noise was the most detrimental, though statistical analysis of performance refuted this belief. We interpret the results and draw insights for system design.

Index Terms: H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems—Artificial, augmented, and virtual realities H.5.2 [Information Interfaces and Presentation]: User Interfaces—Evaluation/Methodology; H.1.2 [Models and Principles]: User/Machine Systems—Human factors

1 INTRODUCTION

Registration error, the misalignment of graphical elements with respect to the projection of the real world in the final image the user sees, has been one of the persistent problems limiting the usability of many augmented reality (AR) systems. Applications vary widely in their need for registration accuracy and the constraints that may be placed on the application or environment in order to achieve it. For example, a medical application requires extremely high accuracy, but the designers may assert control over the environment in order to provide favorable conditions for the component systems to perform well enough to yield this accuracy. On the other hand, an outdoor AR game system may be able to make few assumptions about its environment, but users will likely accept moderate errors in registration so long as they do not interfere with the understanding of the game scenario.

Many AR systems have been built with the assumption that perfect registration will eventually be achieved through some combination of component subsystems. We wanted to determine the need for registration accuracy in our applications [12]. The application examined in this work was situation awareness; that is, providing the user with an understanding of past, current, and potential future events in the surrounding environment. We have typically assumed that perfect registration would be achieved, and thus have often drawn geometric structures in great detail, only to have them wander far from the proper locations (Figure 1).

*E-mail: mark.livingston@nrl.navy.mil

†E-mail: zai@ait.nrl.navy.mil



Figure 1: A typical registration error, in which the rectangles that represent the windows, doors, and the building outline are clearly displaced from the correct locations. The low contrast of this image is due in large part to the difficulty of taking a photograph through an optical see-through display. This and similar images in Figures 2, 3, and 4 have been enhanced to improve the visibility of the graphics.

There have been attempts to reduce registration error through improved tracking and calibration algorithms [1, 2, 19] and intelligent architectures [5, 9]. Still the problem persists for many AR implementations. Furthermore, for open-loop systems, which includes AR systems that use optical see-through displays, there is simply no way to eliminate registration error. Even video AR systems must often make assumptions or constraints in order to enable precise alignment of the graphics [11]. Additionally, scalability limitations, numerical instability, and occlusions could always defeat the tracking.

These practical limitations have prompted a variety of responses by system designers. Perhaps the simplest response, used in many systems, is to design graphics that do not need to be registered with a particular location in the real world. Such 2D overlays can include navigational information such as a map or compass. Another class of demonstrated approaches alters graphical objects to representations that do not require perfect registration, such as blurring graphical models [6] or simplifying or eliminating the geometry [4]. In the latter case, this may include the use of a 2D overlay to express a task to be performed. This places the cognitive load on the user to determine the correspondence between the graphics and the real environment; however, this load is generally considered to be lighter than (or at least no worse than) interpreting misaligned graphics that were drawn under the assumption of accurate registration.

In a typical AR system, there are many sources of registration error [8]. In building up to our goal of analyzing the effect of registration error, we listed the following categories of registration error. This list was not designed to match any theoretical taxonomy of sources; to the contrary, it was designed to match with specifications typically given for commercial tracking systems.

1. *Noise* is perhaps the most obvious dynamic error; it causes the

graphics to jitter around the correct location on the augmented image. This is frequently due to electrical or numerical noise in the tracking system or repetitive interference with the tracking system.

2. *Latency* is a specific type of dynamic error in position or orientation that comes from delay between the time a movement that is being tracked occurs and the system can render the appropriate image in response to that movement. This may be due to sensing or processing delays in the tracking system or resource constraints in the application itself. Depending on the task and technology, latency may be the largest source of registration error [8].
3. *Position error* may be static or dynamic. Static errors may come from system calibration errors, modeling errors, or errors in the tracking system. (The last two will contribute to the first.) Dynamic errors may come from interference in the tracking system (e.g. metallic objects distorting a magnetic field, loss of line-of-sight in an optical tracking system) or other operational characteristics such as poor performance at a particular distance from a tracking source.
4. *Orientation error* may also be static and dynamic, and the two subtypes have similar sources to those of position error.

This list of categories had the nice property that they were all quite easy to systematically vary within an AR system, so that their effects could be studied. Hardware and software sources would not have separated into completely distinct categories. For example, multi-path errors with a GPS receiver could create an error that would belong in the category of position error or in the category of noise, depending on the environment and system configuration. This list lent itself well to a study design that would enable us to specify operating characteristics required from a tracking system, and thus seemed appropriate for our work.

2 RELATED WORK

The difficulty of achieving quality registration gives rise to a number of possible responses to registration error. As noted above, latency is potentially a large source of registration error in many AR systems. Thus it was one of the first to be studied for its effect [14]. Increasing latency was found to create a linear decay in user performance on a matching the distance of a real object to that of a virtual object in a desktop application (maximum distance of 113 cm). Latency was also linearly related to the lateral displacement in the virtual target location as subjects used motion parallax to aid their depth perception. Similar results have been demonstrated for a placement task under latency in virtual environments [20].

In many AR systems, the user is simply expected to adapt to the latency. Under small amounts of latency, this is quite reasonable. A just noticeable difference (JND) of 15 ms of latency was found for virtual environments with both simple and complex scenes [13]. But the JND for AR, in which the reference for the latency is more closely coupled to the senses, could be lower. In a tapping task between visual and haptic feedback, users adapted to degraded registration of the view with the haptic feedback [18], showing no statistical difference between correct camera-to-image-plane distance and 0.5 cm and 2 cm errors in this distance. Sensorimotor adaptation appeared to occur quite rapidly. All three conditions were significantly worse than subjects' performance of the same task in the real world, however.

Position and orientation errors lead to more predictable registration errors. In a target cueing task [21], subjects were faster at detecting virtual targets with precise head orientation for rendering virtual cues (up to 7.5° from target center) and got progressively slower with partially degraded ($7.5^\circ - 22.5^\circ$) and poor

($22.5^\circ - 45^\circ$) cue precision, with the latter condition failing to achieve a significant difference from uncued searching. This effect was greater for low salience (e.g. smaller) targets. Subjects lost trust in the cues as the alignment degraded, which resulted in the *benefit* of wider attentional breadth and more accuracy in finding secondary targets.

Robertson and MacIntyre [15] studied the effect of 2D position errors resulting in registration errors on a desktop assembly application. They divided error into three cases: no error (a baseline case), fixed error (constant direction of position error of one unit in the assembly mechanism), and random error (random direction offset for a position error of one unit). The user was shown a graphical cue for where to place an object, subject to one of these three cases of error. In half the trials, another graphical object that corresponded to a real object was shown, giving the user a context for the error in that trial. They found that users were most accurate in the case of no registration error the least accurate in the case of random error direction. The contextual cue to the registration error improved the accuracy but did not make users faster. Users gained confidence as they learned to adapt without context, but had confidence from the beginning in the presence of contextual cues. A second study [16] found that registered AR yielded faster performance than a head-up display or an AR display situated to the side of the work area; accuracy was not significantly improved in this task.

3 MEASUREMENT OF ERROR

We embedded the control system for this experiment within our AR application for situation awareness [12], in which tracking a target is a potential task that would be asked of a user. Our system is designed for mobile personnel, either on foot or in vehicles, thus we designed our task around tracking the location of vehicles through an environment in which line-of-sight contact is not maintained. As a prelude to determining appropriate variables and values for our experiment, we wanted to measure the four types of errors experienced in our system. We wanted to assess the expected state of the system and the worst-case scenario. For our indoor development station and for this study, we tracked the user's head with an InterSense IS-900 6 degree-of-freedom tracker. We set the sensitivity to 3 (default) and the enhancement mode to 2 (drift correction made smoothly, recommended for HMD tracking). Since this was our first experiment on this subject, we chose to discretize the variables into "high" and "low" errors. For each variable, "low" error was the error typical in the tracking and/or calibration system of our application (described by the measurements below), whereas "high" error included the addition of that type of error.

3.1 Noise

We measured the noise in the IS-900 by placing the sensor on a tripod at the approximate height of a user, allowing the sensor to remain motionless for a few seconds, and then recording the tracker data. We plotted the noise and found its distribution to be consistent with a Gaussian function with a standard deviation in the position and orientation (per axis) up to 1.1 mm in each position axis, 0.12° in yaw, 0.07° in pitch, and 0.05° in roll. This is slightly higher than other reported observations [7]. Since our task was to indicate a direction along the ground plane, yaw was the primary variable of interest. Since the expected value experienced by the user is thus approximately 0.12° , we chose to add 0.24° on top of the system behavior for the "high" case.

We also wanted to include noise in the position estimates of the vehicles. In our application, we measure the position of an outdoor user (dismounted or vehicle) through differential GPS. Consistent with our experience and reported measurements [10], we assumed Gaussian white noise that in the expected state ("low") has a variance of 0.3 m and in the "high" case has a variance of 1.0 m.

3.2 Latency

We assumed the existence of two frames of latency, one in the rendering and one in transfer of the frame buffer to the display. This implied a latency of 33 ms at 60 Hz; consistent with our observations as well as those of others [7], we expect approximately 20 ms of latency from the tracker, for a total of about 50 ms of end-to-end latency. For the worst case, we chose 150 ms. This was applied to the vehicle positions, but no additional latency was added to the user's orientation. We decided not to do the latter due to concerns about causing dizziness or frustration on the part of the user. Prediction over short intervals has been demonstrated to improve AR registration [1]. The IS-900 has a prediction mechanism of up to 50 ms, which is applied to its position and orientation estimates.

3.3 Orientation Error

In order to measure the orientation error, we captured images through the optics of the display. While there is no guarantee that the camera center was at the same position as any user's eye, this should give a good approximation to the orientation error the user experienced. We placed the camera up to the display and performed our standard calibration procedure (described in Section 4.2) using the camera's LCD for feedback. We then turned the display and camera assembly approximately 45° to the left (i.e. in yaw), and then back. This turn creates "new" information for the extended Kalman filter (EKF) embedded in the tracking system and causes any settling in the tracking and calibration to occur. (This turn is not truly necessary for the error to occur, but some motion is needed so that the EKF does not ignore measurements from a stationary receiver. Also, turning to the right has not been shown to make any difference in the error, nor has the error been shown to increase over time or with more turns.) We waited a few seconds after this turn to capture images, so that latency was not a concern. We inspected the captured images (Figure 2 left) for the distance in pixels between a window edge and the corresponding (graphical) line. Using the known measurements of the window and the distance to it, we determined the angular error for that image. All of the pixel error was attributed to orientation error, which comes from either the tracker or system calibration; this is a suitable approximation because at the distance of the building (61 m), orientation error dominates the registration error for the stationary HMD [3].

We captured 18 such images, resulting in a mean error of -0.2° with a standard deviation of 1.2° . (The negative sign indicates that the error placed the graphics to the left of the real environment.) The range of errors was -2.3° to 1.5° ; we thus set the additional orientation error to -2.4° to create a worst-case scenario. This means that we might expect a user to experience anywhere from -2.3° to 1.5° of orientation in the expected case and between -4.7° to -0.9° in the worst case. We did not attempt to measure the error for each user, let alone for each task or trial.

3.4 Position Error

We also measured the position error experienced by our users with a similar procedure as for the orientation error. A nearby indoor target cross-hairs (with its virtual analogue) was used, and after calibration, the display and camera assembly was translated to 1.0 m from the physical cross-hairs. We captured 18 images (Figure 2, right), measured the pixel error, and computed the linear error along the ground plane (approximately aligned with the world x -axis). We found a mean of -20 mm with a standard deviation of 6 mm. While this measurement does not account for the effect of the orientation error within the images captured for measurement of position error, we use the observation [3] that when looking at a nearby target, the position error dominates the registration error. Our measured position error is higher than other reported observations [7]; it should be noted that our tracking volume ($10.4 \times 5.3 \times 3.0$ m) is significantly greater than in the other experiment, however. Also, according to

the above calculations, we may be attributing (on average) 3.5 mm of registration error to position when it is in fact due to orientation error ($\tan^{-1}(0.2) = 0.0035$).

4 EXPERIMENTAL DESIGN

Our current hardware implementation uses an nVisorST optical see-through display (1280×1024 resolution, 40° vertical field of view). The housing enables each of the left and right displays to be moved independently side-to-side for centering on the user's eyes. The image generation platform was a 3.06 GHz Pentium4 with an NVIDIA Quadro4 900XGL. The nVisorST display requires two 1280×1024 images at 60 Hz; we generated these as a single stereo image and split the left and right halves into two video streams for input to the display. We did not study stereo as a variable, on the assumption that it would have no effect for distant objects.

We balanced the amount of light from the real environment against the brightness of the graphics by layering neutral density filters in front of the display. With bright (whitewashed) buildings in our environment, this was necessary so that the brightness of the real world did not overwhelm the brightness of the graphics. We found that three layers of the 0.3 density filter (1 f-stop, 50% transmission) were sufficient for normal light levels experienced at our lab doorway. Two layers were used for a few subjects who completed the experiment on rainy days. Two subjects found that three layers were insufficient for periods of intense sunshine; they used their hands or the door frame to completely block the incident light and see the graphics.

4.1 Experimental Task

We asked users to follow a target vehicle model, which differed in shape and color from two distractor vehicles (Figure 3). Associated with each vehicle was a white box that is used as a cue for the location of moving entities in our application [12]. We used virtual models as a proxy for real objects so that we could control the behavior and have their actions repeat for each subject. The virtual vehicles thus disappeared behind buildings, whereas the box that cued the user to the vehicle locations did not; this box behaved as the graphics would in our application, in that it was visible even though it may have been geometrically occluded from the user's view.

Upon completion of the calibration done for each task, the target and distractor vehicles were introduced into the environment. The user stood in a location that maximized the ground area directly visible outside our building through a set of doors and windows and hit a key to begin the task. At pre-determined locations of the target, all vehicles and their augmenting boxes froze in world position. (Any latency errors in the boxes' locations were thus also frozen.) Prediction and noise continued to affect the tracker measurement given to the renderer, as did any additional noise error we introduced as an independent variable. (See below.) When the target froze, a cross-hairs appeared in the center of the display in both eyes (Figure 4), and the user was asked to align the cross-hairs with the target vehicle (explicitly not with the associated box, even though the vehicle may not have been visible). To indicate that the cross-hairs was aligned with the vehicle, the user hit a key.

We strongly suggested to users that they actively follow the target, though there was no requirement that they do so. However, this was the only method by which they could know that they were correctly following an occluded target. Some users did check whether they had lost track of the target by searching the area for any visible vehicles. Users did not try to achieve high accuracy until the target froze and they were asked to be accurate.

Upon identifying the target direction, the user was also asked to call out the number of occluding buildings between his/her location and the target vehicle; this number was 0 (directly visible or level-0 target), 1 (one occluding building, which would itself be directly

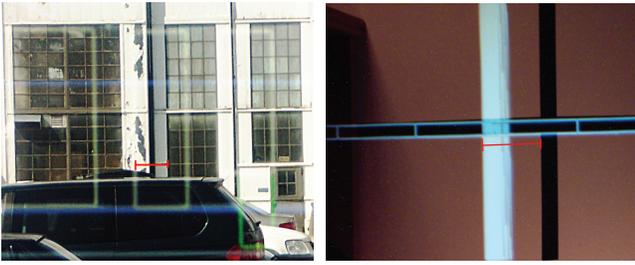


Figure 2: Images from the measurement of error in orientation (left) and position (right). We computed angular error from the error in pixels (red bar near the center of the image), the width of the window, and the distance to it. Similarly, we computed position error in the ground plane from the horizontal displacement between a real and virtual cross-hairs and the distance.

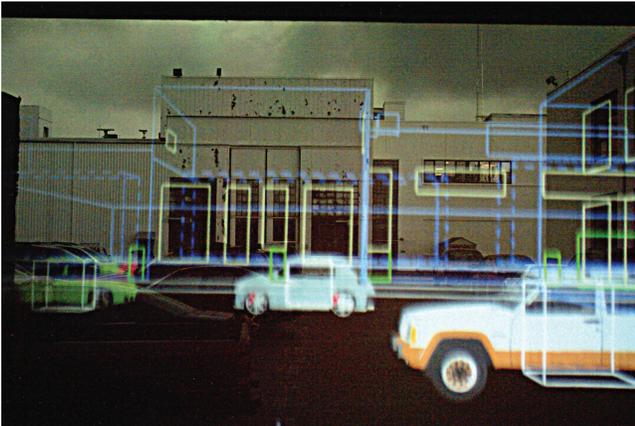


Figure 3: This image shows the target vehicle (gray hatchback, center) and the two distractor vehicles, along with the geometric model used for calibration purposes and for the task of identifying the number of occluding buildings.

visible, a level-1 target), or 2 (two occluding buildings, the second of which was only visible in the graphics, a level-2 target). During a training task, the user was talked through this procedure and informed of the possible correct answers to the number of intervening buildings (including part of the building in which the experiment was run, not counting the windows where the user stood). All users should have been aware of the layout of the buildings; some reported using this knowledge to respond.

We designed six sets of routes (Figure 5); each set contained a route for the target and a route for each of two distractors. Each route for the target had six points at which all three vehicles would stop along with their augmenting boxes. Each route set was designed to have two cases of visible targets, two cases of occlusion by one building or part of a building, and two cases of occlusion by two buildings. (The distractors and their visibility were not controlled.) However, due to imprecision in the map used to plan the routes and movement of the users during the experiment, some deviations occurred. Also, there were situations in which a distractor vehicle could have occluded the target, and although users were instructed to consider only the buildings, we believe some considered the distractor vehicles. Two routes turned an intended level-2 target into a level-1 target; one converted a level-2 into a level-0, and one changed a level-1 into a level-2. Thus over all six routes, there were 13 level-0 targets, 13 level-1 targets, and 10 level-2 targets, ignoring potential occlusions of the target by distractors.

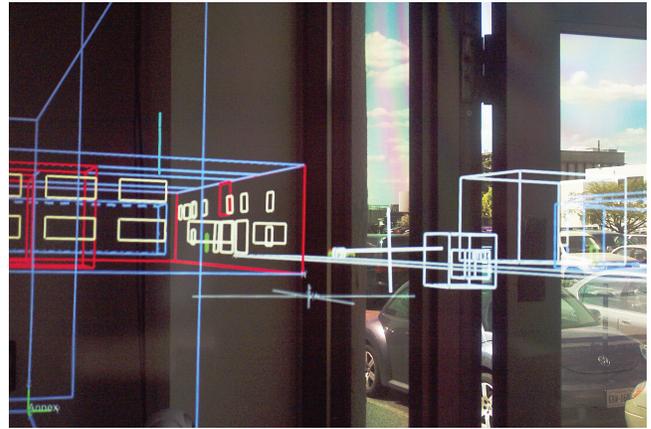


Figure 4: Users identified the direction to the target by aligning the virtual cross-hairs with the 2D projection of the target's location. In the case of a visible target, this was indicated by the target model. In the case of an occluded target, this was indicated by the box, except that the box exhibited registration error.

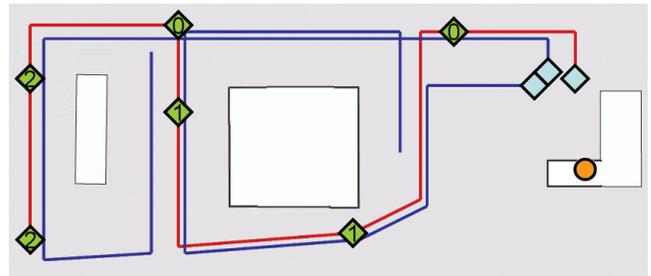


Figure 5: Routes wound between and behind buildings. The target (red) and distractor (blue) paths ensured that all three vehicles were visible at the start (light blue) of each task from the user position (orange). The stopping points (green) for the target indicate the number of intervening buildings that the user should identify. Note that the user stood in a windowed doorway, and thus had a large field of regard for unobstructed viewing.

4.2 Subject Procedures

Each user completed a general questionnaire; we then measured the user's IPD (for rendering) and height (for calibration). The user then donned the display and used a series of concentric circles [17] to center the display over each eye. We then presented a calibration image that has become standard for our system to determine the offset of the yaw angle between the tracker's compass and the world. We used this image for two calibrations. Before the user began any tasks, the user was asked to verify the sizes of a set of virtual rectangles against real windows and doors (visible in Figure 1) in order to check that the field of view experienced was matched by the rendering. The experimenter would have operated a scrollbar to make any necessary adjustments, but no user requested any. At the start of each task, we fixed the image relative to the user's head. The user was asked to turn his or her head until that set of windows aligned with the real environment. The offset between the tracker frame and the rendering frame of reference was recorded. After all tasks, users completed a subjective evaluation.

Twelve subjects each completed twelve tasks with six trials, for a total of $12 \times 12 \times 6 = 864$ trials. One subject withdrew due to dizziness (a possibility that the subject noted would be likely due to a medical condition), and one subject withdrew due to extreme difficulty completing the task (likely due to color-blindness and depth

| Vehicle position | | | | | |
|------------------|-------------|-------|--------------|-------|---------------|
| Variable | “Low” value | | “High” value | | Predict Added |
| | System | Added | System | Added | |
| Noise | 0.0m | 0.3m | 0.0m | 1.0m | |
| Latency | 0ms | 0ms | 0ms | 150ms | |

| Head Orientation | | | | | |
|------------------|-------------|-------|--------------|-------|---------------|
| Variable | “Low” value | | “High” value | | Predict Added |
| | System | Added | System | Added | |
| Noise | 0.12° | 0° | 0.12° | 0.24° | |
| Latency | 50ms | 0ms | 50ms | 0ms | -50ms |
| Ori Error | -0.2° | 0° | -0.2° | -2.4° | |

Table 1: Independent variables for error types and the values used. The System column gives our measured value for the given type of error, whereas the Added column indicates the additional amount added for the two or three discretized levels of error. Prediction is only used for head orientation.

perception difficulties). Of the twelve subjects who completed the task, eleven were male. Subjects ranged in age from 25 to 58, with a mean of 34.3 years. All reported being heavy computer users with good or excellent spatial visualization capabilities; seven reported being frequent players of computer games.

4.3 Independent Variables

Given the distance between the user and the target and to reduce the number of variables being studied (and users required), we chose noise, latency, and orientation error as our independent variables. The noise values we measured were used as the standard deviation of a Gaussian distribution from which the noise was selected in each frame. This orientation noise error was then added to the reported head orientation, and the position noise error was added to the vehicles’ positions. Similarly, the latency we estimated and orientation error we measured existed, and we applied additional amounts. We used the IS-900 prediction mechanism set to 50 ms as a third discretized value for latency; it was applied to the user’s head position and orientation. We did not create any error in the orientation of the vehicles, under the assumption that this would have little effect on any aspect of the task performance. Table 1 summarizes the independent variables we used in the experiment and their values as measured or introduced for our experiment (Section 3).

4.4 Dependent Variables

We recorded the position and orientation of the user’s head when the direction to the target vehicle was specified; we also recorded the location of the target vehicle and the two distractors. From the positions, we computed the actual angle to the target and each of the distractors. With the user’s orientation, we computed the response for the angle. The difference between these two gave us one measure of error. We recorded the time the user took to enter this response, as well as the correct and responded number of intervening buildings at each trial. The difference in the last two gave us another measure of error. Also, using the locations of the target and the distractors, we could determine whether the user followed the target or one of the distractors. Finally, we verbally and visually explained the three types of errors and asked to users rate the difficulty presented by each on a ten-point Likert scale. (Users were able to properly repeat the definitions of the types of error in their own words, so that we knew the definitions were clear.) This subjective assessment was done once, after completing the set of twelve tasks. We also asked users to indicate signs of general discomfort (dizziness, eye strain, fatigue) both before and after the experimental session.

4.5 Counterbalancing

We used a $2 \times 3 \times 2$ Latin squares design for counterbalancing of order effects between the noise, latency, and orientation error variables. Due to a transcription error, the actual design was slightly unbalanced; one case of each level of orientation error was missing for low noise and prediction on for the latency. In order to reduce the programming, we repeated each task twice per subject (with different values for the independent variables). We used a random permutation of the six route sets on the first six tasks a user saw, and then a second random permutation for the last six tasks, under the condition that neither of the last two route sets from the first half were used in the first two tasks from the last half. This reduced the chance that a user would recognize a set of routes and answer from memory rather than the system’s behavior. We further selected the set of random permutations over all users to counterbalance order effects of the sets of routes.

4.6 Hypotheses

We expected the following outcomes for the independent variables.

1. High noise would have the most detrimental effect on the users’ ability to follow the correct target and would reduce the precision with which they localized the target.
2. High latency would have a detrimental effect on the users’ precision in localizing the target but little effect on their ability to follow the correct target. Prediction would improve their localization, but have no effect on their ability to follow the correct target.
3. High orientation error would have a detrimental effect on the users’ precision in localizing the target, approximately equal to the amount of error introduced (though the uncontrolled system error varies widely and may affect the localization as well). It would have little effect on the users’ ability to follow the correct target.

In addition, we knew that certain route sets would be more difficult than others, such as when the distractors were near the target, especially for extended periods of time.

5 RESULTS

We conducted several analyses to determine statistically significant effects for the variables described above. All the reported numbers were computed using factorial analysis of variance (ANOVA) with SYSTAT 11.

5.1 Analysis of All Data

5.1.1 Following the Target

Perhaps the most fundamental measure is whether the user was able to correctly follow the target. This was quite difficult in certain tasks or under certain conditions of the independent variables. We used the following Boolean expression to determine whether the user was following the target rather than either of the distractors.

$$\left\{ \begin{array}{l} |\alpha_t - \alpha_R| \leq |\alpha_{d_i} - \alpha_R| \text{ OR} \\ (|\alpha_{d_i} - \alpha_R| < |\alpha_t - \alpha_R| \ \& \ |\alpha_t - \alpha_{d_i}| < 10^\circ) \end{array} \right\} \text{ for } i = 1, 2$$

where α denotes an angle within the world ground plane for the target (subscript t), the distractors (d_i for $i = 1, 2$), or the user response (R). Essentially, either the user’s indicated direction was closest to the target direction, or it was closest to a distractor that was, at that stop, under 10° from the target. If this expression was true, then the user was considered to have been correctly following the target.

We ran a $2 \times 3 \times 2$ within-subjects, repeated-measures ANOVA. There were, surprisingly, no main effects of any of the independent variables for the users’ ability to correctly follow the target.

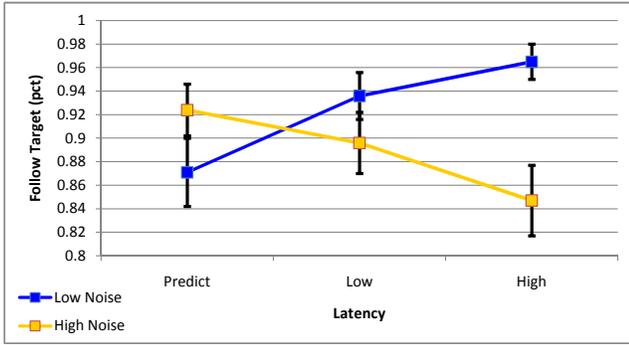


Figure 6: The interaction of noise and latency on correctly following the target (higher percentage indicates better performance) showed that while low noise was generally better, prediction seemed to cancel the negative effect of high noise. In this and all such graphs, the error bars indicate one standard error.

Noise and latency had a significant interaction, $F(2, 22) = 6.222$, $p \approx 0.002$, for the users' ability to correctly follow the target (Figure 6). The prediction mechanism available in the IS-900 appeared to help overcome the negative effect of higher noise. One possible explanation for this is that the prediction equations, which are based on classical mechanics, helped smooth the noise from the tracker (but obviously not the noise introduced by our stimulus generation code), and thus lowered the total noise experienced in that state. It is perhaps counter-intuitive that performance was lower with prediction on and low noise than with prediction on and high noise. However, this difference is not statistically significant; using the Welch-Satterthwaite equation, $t(250) = 1.249$, $p \approx 0.106$.

Noise and orientation error also had a significant interaction, $F(1, 11) = 7.270$, $p \approx 0.007$, for the users' ability to correctly follow the target (Figure 7). High orientation error seemed to somehow cancel the disadvantage that high noise gave to the user. Again, that high noise allowed a better performance than low noise under any set of values for the other variables is counterintuitive, but this difference was not significant: $t(214) = 0.5235$, $p \approx 0.297$. This effect may also explain a trend, $F(1, 11) = 3.252$, $p \approx 0.072$, for higher noise to improve performance. Based on these two interactions and some user comments, we believe high noise may have helped users differentiate a target from a distractor in certain situations. Further analysis of the routes may help us understand if this trend may be of interest or it may help determine the suitability of a set of routes.

5.1.2 Response Time

There were main effects for noise and latency on response time (Figure 8). Recall that these conditions have a fundamental difference: noise in the user's position and orientation that came directly from the tracker continued while the user was aligning the cross-hairs to the (presumed) target. Additional noise introduced to the user's head orientation for purposes of this study also continued. But noise in the vehicle positions did not, and the latency that existed at the time the vehicle froze for the user to perform the localization task did not disappear from the location of the box associated with the vehicle. Users were about 10% faster with low noise, $F(1, 11) = 4.234$, $p \approx 0.037$.

The overall effect of latency was for users to be faster with low latency than prediction, and faster with prediction than high latency, $F(2, 22) = 4.061$, $p \approx 0.018$. But the pairwise tests between low latency and prediction - $t(574) = 1.508$, $p \approx 0.066$ - and prediction and high latency - $t(443) = 1.548$, $p \approx 0.061$ - are merely

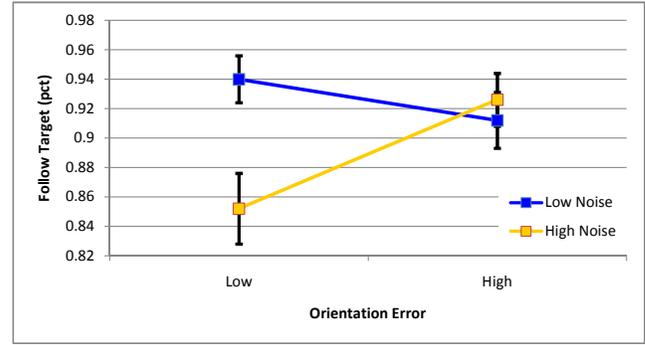


Figure 7: The interaction of noise and orientation error showed an unexpected improvement under high orientation error and high noise.

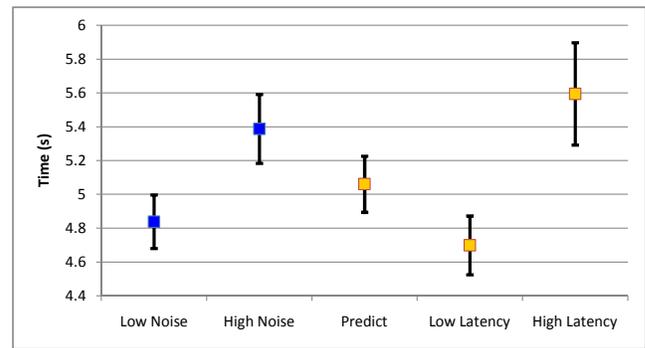


Figure 8: The response time had main effects from noise (blue) and latency (red). Subjects were faster with low noise and low latency. The former is not surprising, given that the user's orientation was subject to jitter with high noise. The latter would indicate that users were trying to think about the effect that latency had at the moment the graphics froze and awaited their response.

trends. Each step added about 10% to the time required. Also, noise and orientation error had an interaction for response time; similarly to above, high orientation error seems to cancel the benefit of low noise, $F(1, 11) = 3.969$, $p \approx 0.047$.

5.1.3 Occluder Level

Noise and latency had a significant interaction, $F(2, 22) = 3.871$, $p \approx 0.022$, for identifying the level - i.e. the number of intervening objects from the user's location to the identified target. Similarly to the result above for following the target (and perhaps because of the lower error rate), users were more accurate in identifying the number of occluding buildings under high noise than under low noise when prediction was on. For low or high latency, however, users were more accurate when the noise was lower (Figure 9). It could be that prediction smoothed the noise enough to help the user in this aspect of the task as well. When restricted to inlier data (below), the difference between the high noise, low latency condition and the low noise, low latency condition was not significant, $t(277) = 0.424$, $p \approx 0.336$.

5.2 Analysis with Outliers Removed

We labeled any trial for which the user was not following the correct target (as determined by the Boolean function above) as an outlier, then re-ran the analysis. We performed a $2 \times 3 \times 2 \times 3$ within-

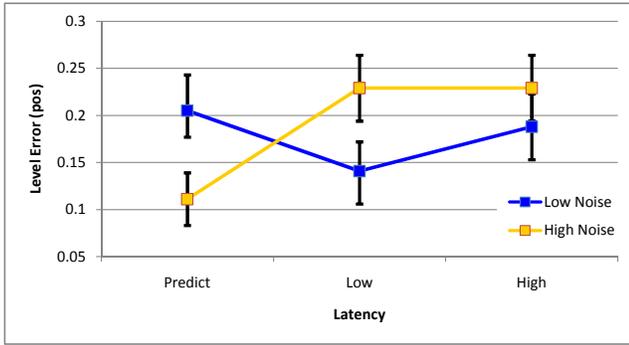


Figure 9: The interaction of noise and latency on assessing the number of occluding buildings showed that users performed better under high noise than low noise when prediction was enabled. Units are positions in the order.

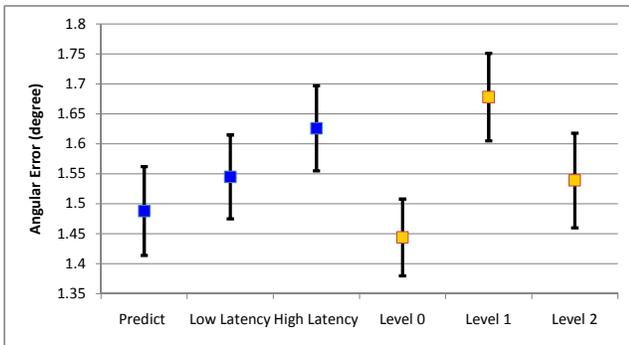


Figure 10: Latency and level (number of occluders) both had a main effect on angular error when outliers were removed. Users showed a small but statistically significant improvement with reduced latency and with prediction. Users showed an expected performance benefit in localizing visible targets

subjects, repeated-measures ANOVA. The fourth variable in this analysis is the actual number of occluding buildings between the user and the target. The design is not balanced with respect to this variable because we allowed users to move position. Targets that were near corners may have been at a level other than the initial design assumed it would be. The analysis uses the subject position on each trial to compute the correct number of occluding buildings.

5.2.1 Angular Error

Latency had a main effect for the angular error in localizing the direction to the target, $F(2, 22) = 4.319$, $p \approx 0.014$. Users appeared to treat the location of the box as correct, even though they were cautioned that it would be incorrect in many cases and this was demonstrated in the early part of every task, when both the vehicles and their associated boxes were present. Users did not appear to reduce the registration error of the box in their minds when they localized the target. Thus their performance was best with prediction on, in the middle with low latency, and worst with high latency (Figure 10). But these differences, though significant, were so small as to be unlikely to be meaningful in our application, under 0.1° from prediction to low latency and from low latency to high latency.

The number of occluding buildings also had a main effect for angular error. Not surprisingly, users were most accurate for visi-

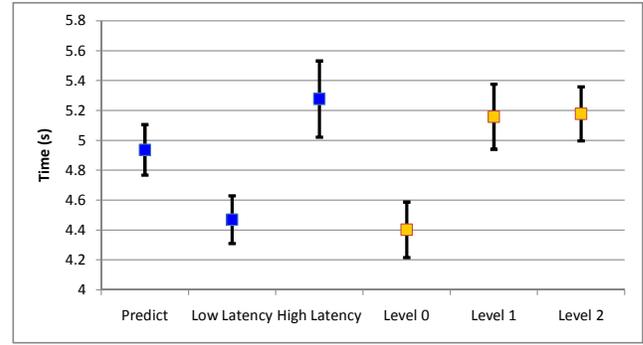


Figure 11: Latency and the level of occlusion both had a main effect on the response time. Subjects were fastest under low latency, appearing to think about the implications of the prediction and high latency conditions. Subjects were faster when they could directly see the target. Some of this time could be due to thinking about the occlusion level and not the localization task.

ble targets, $F(2, 22) = 82.669$, $p \approx 0.000$ (Figure 10). Strangely, however, subjects were better with level-2 targets than with level-1 targets, but this difference was not significant (Welch-Satterthwaite: $t(445) = 0.940$, $p \approx 0.174$). It may be that since occluding buildings often left nothing behind the target users localized level-2 targets better, though. We have not analyzed the cases where the user succeeded on the level-2 targets, and again it is unclear that the magnitude of the performance difference (up to 0.25°) would have any meaningful effect in our application.

5.2.2 Response Time

Latency had a main effect for response time, $F(2, 22) = 3.481$, $p \approx 0.031$. Users were fastest with low latency, about 10% slower with prediction on, and an additional 7% slower with high latency (Figure 11). So it appears that users were indeed trying to think about the effect of the latency condition, even though it did not improve their performance in localizing the target.

The number of occluding objects also had a main effect on the time, $F(2, 22) = 5.035$, $p \approx 0.007$ (Figure 11). As expected, users were much faster localizing visible targets, and about 17% slower for occluded targets (with no significant difference between one and two occluders). However, it should be noted that we did not separate the time for the localization task and the identification of the number of occluders, so it is possible that the slower time is due to the level identification rather than difficulty of the localization task.

There was also a significant interaction between noise and latency for time (Figure 12). High latency appeared to defeat the benefit derived from low noise, $F(2, 22) = 3.016$, $p \approx 0.050$. This would imply that subjects were trying to consider the effect that the high latency had on the position of the vehicle.

5.2.3 Occluder Level

We asked subjects to identify the number of occluding buildings each time they localized a target. It should be no surprise that the actual number of occluders had a main effect on their ability to perform this part of the task, $F(2, 22) = 22.014$, $p \approx 0.000$. Subjects were far worse in identifying cases in which there were two occluders than in cases of either one occluder or no occluders (Figure 13). Users were slightly more accurate with one occluder intervening than with visible targets. This difference was only a trend (using Welch-Satterthwaite, $t(554) = 1.41827$, $p = 0.078$). This may indicate a flaw in the Boolean expression for determining outliers or

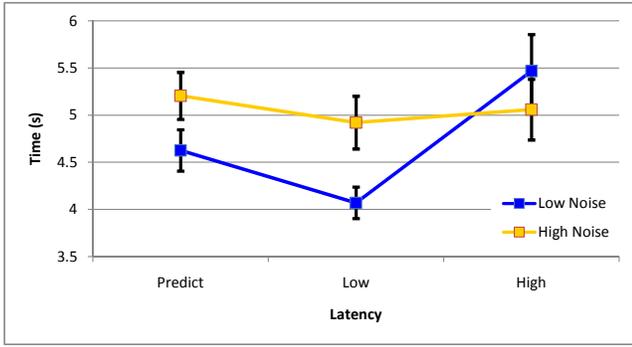


Figure 12: The interaction between noise and latency for response time showed that high latency appeared to defeat the benefit of lower noise. This would imply that subjects were thinking about the effect of the latency.

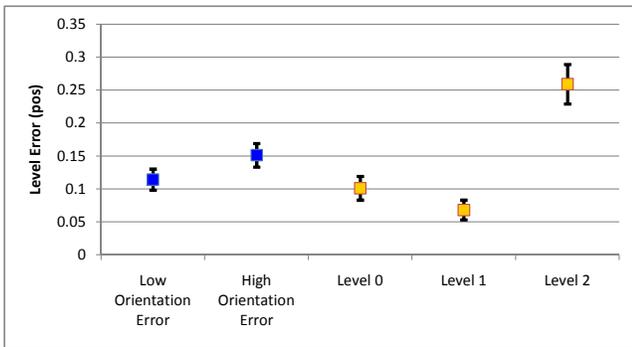


Figure 13: The target level had a main effect on the error in identifying the number of occluders. Surprisingly, there was no significant difference between level-0 and level-1 targets; users had difficulty with level-2 targets. Orientation error showed a trend; we suspect an unmeasured interaction with the structure of the environment.

demonstrate a need for better routes. Or perhaps a greater number of trials would reverse this trend.

There was also a trend for orientation error on the number of occluders (Figure 13). As one would expect generally, users were better with low orientation error, $F(1, 11) = 3.455$, $p \approx 0.063$, although why this would affect the understanding of the number of intervening buildings is unclear. It was possible that the exact locations within the environment had some effect on users' understanding of the target location relative to the environment. This may also speak to the sensitivity of this task to the location within the environment of the building locations.

5.3 Analysis of Weather

We ran one final analysis to look for significant effects with the weather outside. This is an important question for our system, because the intended application (hardware and software) must be robust enough to work outdoors in any weather conditions. Also, our campus features white buildings which, with sunshine, can become extremely bright relative to their surroundings. We were fortunate in that six subjects completed the experiment in bright sunshine, three in cloudy conditions, and three in rainy conditions. This gave us hope that there would be sufficient data to find significance from the weather. We ran a $2 \times 3 \times 2 \times 3$ mixed-design, repeated measures

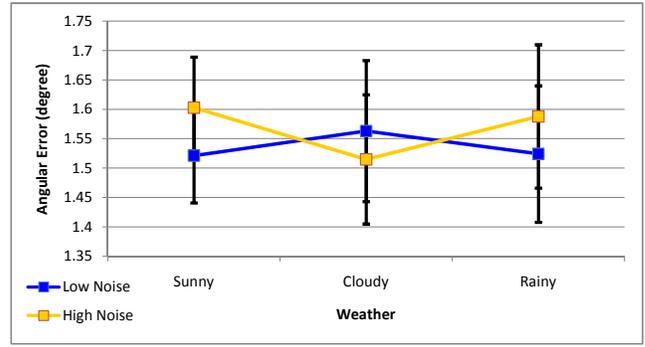


Figure 14: The interaction of noise in the tracker and the outside weather on the localization task.

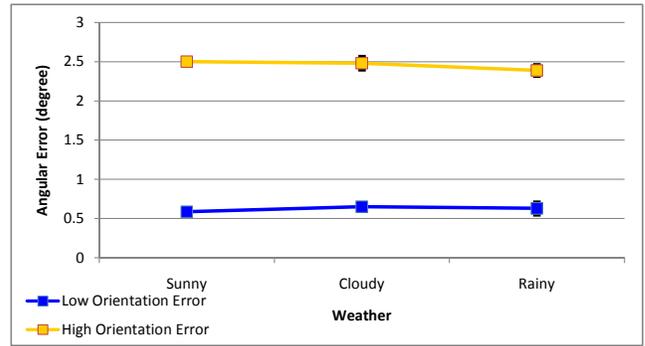


Figure 15: Despite the high difference in error, there was a significant interaction between the orientation error and the weather.

ANOVA; the weather is a between-subjects variable, while all other variables are within-subjects and the repetitions of tasks and trials are as described above. There were no main effects from the weather. We report here only those effects that involve the weather as a variable; the effects reported above appeared again in substantially the same values and are not repeated.

5.3.1 Angular Error

Noise and weather showed an interaction for angular error in localizing the target, $F(2, 22) = 3.073$, $p \approx 0.047$. We would have expected that low noise would have been better in any weather, but under cloudy conditions, users were more accurate under high noise (Figure 14). These differences are likely not meaningful for our application, however, as the differences in the average are under 0.1° . Still, it raises the question of the importance of the brightness problem (due to sunshine) relative to the tracker errors introduced for our study.

Orientation error and weather also had an interaction for angular error, $F(2, 22) = 5.934$, $p \approx 0.003$. Users were clearly better – by somewhat less than the additional error introduced – for the low error condition than the high error condition. But in the high error condition, users were getting better as the skies got darker, whereas in the low error condition, users were best in sunny weather, and slightly worse in both the cloudy and rainy conditions (Figure 15).

5.3.2 Time

Noise and weather had an interaction for time, $F(2, 22) = 3.005$, $p \approx 0.050$. With low noise, users were slowest with cloudy con-

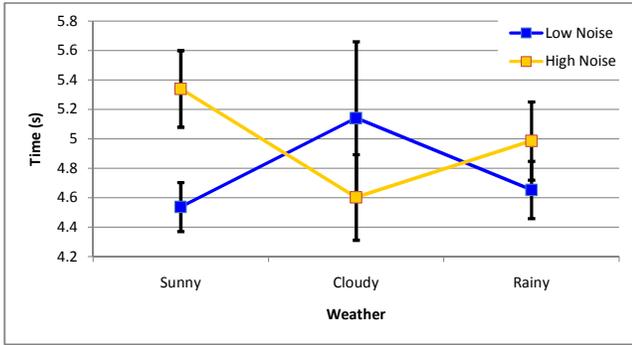


Figure 16: Noise and weather had an interaction on the response time. Users were slowest with high noise and sunny weather, but fastest with low noise and sunny weather.

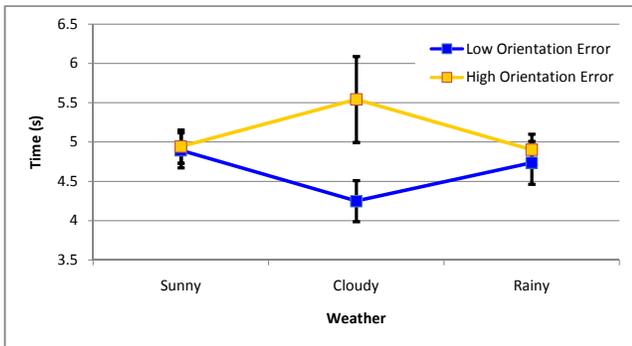


Figure 17: Orientation error and weather had an interaction for response time. Under cloudy weather (perhaps the ideal viewing condition), there was a difference between high error and low error, but not for bright sunny skies or the very dark rainy conditions.

ditions, whereas with high noise, users were fastest with cloudy conditions (Figure 16). We would have thought, given the brightness difficulties posed by the sunlight reflecting off the buildings, that users would have been slowest in sunny conditions as they adjusted their view to make the graphics visible. But it appeared that the combined effect of high noise and bright sunshine slowed users down even more.

Similarly, orientation and weather had a trend for time, $F(2, 22) = 2.804$, $p \approx 0.061$. There was no difference due to orientation error for sunny or rainy conditions. But under cloudy conditions, users were somewhat faster with low orientation error than with high orientation error (Figure 17). This likely speaks to the difficulties in adjusting the relative brightness of the display to the sunny conditions (which can overwhelm the display) and the rainy conditions (which were quite dark with the density filters in place).

5.4 Subjective Results

Users believed that noise was the most detrimental variable on task performance (Figure 18); the difference was quite emphatic, $t(22) = 5.252$, $p \approx 0.000$ between noise and latency. Of the twelve subjects who completed the experiment, four reported an increase in eye strain and six reported being quite fatigued. The weight of the nVisorST is balanced on the user’s head, but the users’ reactions reflected the lack of comfort with wearing the bulk of the display. These results are typical for our experiments. Two subjects had

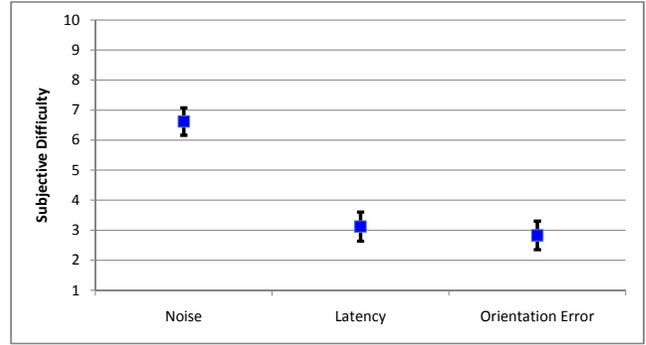


Figure 18: Users’ subjective ratings on a ten-point Likert scale showed their belief that noise was the most detrimental of the three types of error; the difference in between noise and the other two types of errors was significant, even though the performance measures did not show a significant difference in main effects.

experience wearing AR displays, but not with the nVisorST.

6 CONCLUSIONS

We expected noise to have a strong negative effect, but we did not find a large impact on performance. It slowed users down (by small percentages) and was displeasing in a subjective sense, however. Thus we can not accept our first hypothesis, that noise would have the strongest negative effect on the users’ ability to follow and to localize the target. But it does seem safe to assert that high noise exerted a negative influence on users, if not at the level or in the manner that we expected. It appears that prediction to combat tracking latency had a positive side effect of smoothing out the noise, which was not an expected result, but certainly has implications for the tracking systems and applications that use them.

Latency exhibited a similar behavior to noise on the task of following the target and on response time. When we removed outliers from the analysis, latency was shown to have had a significant effect on the localization of the target; thus we can accept our second hypothesis. Users appeared to simply fix their attention on the box and ignore any extrapolation that would have enabled them to overcome the registration error caused by latency. As expert AR users, we are accustomed to doing this; though we instructed our subjects to adjust, they did not. Users were slower under prediction and under high latency, compared to working under low latency. It would be easy to explain the delay under high latency as users trying to determine how to compensate for the visible separation, but this would not work as an explanation for the delay in responding under the prediction condition. These results require further investigation.

It was strange to see that orientation error did not have a significant effect on the localization accuracy. This would have seemed to have been the most obvious prediction to make. But we can offer possible explanations. The mean error was approximately 60% of the additional error, and there could have been significant variation in the amount of “low” orientation error. Thus the limited amount of control we were able to exert in this task may have made the orientation error an inconsistently-coded variable. Figure 14 shows a performance gap between low and high orientation error of approximately 1.8° . This is less than the additional error we introduced, and the error at the “high” level was curiously close to the amount we introduced. It would appear wise for us to search for when the various types of error dominate the task performance.

The analysis with respect to weather was unplanned, but nonetheless a fortunate circumstance. We have long observed that bright sunshine presents problems for any optical see-through dis-

play. The neutral density filters reduced the brightness of the real world in the merged view; we needed more than we expected to contend with sunshine and less than we started to use for rainy conditions. This certainly gives us some insight into the requirements for the HMD to be useful in our outdoor application. The interactions of weather with the independent variables was curious. It would make some sense that noise was particularly challenging on a sunny day, when the graphics were lower in salience (with a fixed amount of transmission from the real environment). The jitter may decrease the effective intensity such that more attention is needed to perceive and understand them.

There are thus a number of avenues for future work. Clearly, some aspects of the task were not well-suited for our investigation. One route set was particularly difficult for users; it accounted for 71% of the errors in following the target (many of which occurred in sequence). Also, we should resolve ambiguities raised by occlusions near building boundaries and by the distractor vehicles. The issue of the balance between light from the real environment and from the graphics needs to be resolved in a more automatic way; this is well beyond the scope of our project, but potentially useful feedback for the display designers. For our purposes, we need to take some measurements that will enable us to continue conducting studies in all types of outdoor weather.

We would like to expand the variables studied to include the entire list presented in Section 1. This would likely require adapting a table-top assembly task [15] to our needs in order that each type of registration error has a chance to be significant in the application. It would help to have online measurements of the four errors we categorized; this is unfortunately difficult for optical see-through systems, which are a desired user characteristic for our application. But it does encourage us to consider more detailed characterizations of the error; this may help elucidate details of the effects. We would like to recruit more subjects and a varied population; however, our target users are Marines or soldiers, who are predominantly male and heavy computer game players.

We devised a framework in which we can test the effect of various types of registration error from the user point of view. These types of errors directly relate to standard performance metrics for commercial tracking systems. We found users were surprisingly robust to registration error caused by noise, latency, and orientation error. As previous studies had found, users tolerated latency, although their accuracy on the task decreased. Users reacted negatively to noise, but it did not hurt their performance significantly or as much as they appeared to think it did. Orientation error and latency predictably caused users to simply follow the cue; they did not recognize a pattern of error. The variability of the condition in a real system may contribute to this.

The encouraging news is that we have begun to quantify the effects of various forms of registration error. More precise studies with greater numbers of levels of error will give us the opportunity to give precise requirements for tracking systems in order to make our AR system more usable in the field.

ACKNOWLEDGEMENTS

The authors wish to thank Adam Jones, Marc Foglia, Ron Couvillion, the anonymous subjects, and the reviewers.

REFERENCES

- [1] R. Azuma and G. Bishop. Improving static and dynamic registration in an optical see-through HMD. In *Proceedings of SIGGRAPH'94*, Computer Graphics, Annual Conference Series, pages 197–204, Aug. 1994.
- [2] Y. Baillot, S. J. Julier, D. Brown, and M. A. Livingston. A general tracker calibration framework for augmented reality. In *IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 2003)*, pages 142–150, Oct. 2003.
- [3] M. Bajura. Camera calibration for video see-through head-mounted display. Technical Report TR93-048, Dept. of Computer Science, University of North Carolina at Chapel Hill, July 1993.
- [4] E. M. Coelho, B. MacIntyre, and S. J. Julier. OSGAR: A scenegraph with uncertain transformations. In *IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 2004)*, pages 6–15, Nov. 2005.
- [5] T. Edmunds and D. K. Pai. An event architecture for distributed interactive multisensory rendering. In *IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 2006)*, pages 197–202, Oct. 2006.
- [6] A. Fuhrmann, G. Hesina, F. Faure, and M. Gervautz. Occlusion in collaborative augmented environments. *Computers and Graphics*, 23(6):809–819, Dec. 1999.
- [7] S. J. Gilson, A. W. Fitzgibbon, and A. Glennerster. Quantitative analysis of accuracy of an inertial/acoustic 6dof tracking system. *Journal of Neuroscience Methods*, 154(1–2):175–182, June 2006.
- [8] R. L. Holloway. Registration error analysis for augmented reality. *Presence: Teleoperators and Virtual Environments*, 6(4), Aug. 1997.
- [9] M. C. Jacobs, M. A. Livingston, and A. State. Managing latency in complex augmented reality systems. In *1997 ACM Symposium on Interactive 3D Graphics*, pages 49–54, Apr. 1997.
- [10] C. Kee, B. Park, J. Kim, A. Cleveland, M. Parsons, and D. Wolfe. A guideline to establish DGPS reference station requirements. *Journal of Navigation*, 61:99–114, 2008.
- [11] G. Klein and D. Murray. Parallel tracking and mapping for small ar workspaces. In *IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 2007)*, pages 225–234, Nov. 2007.
- [12] M. A. Livingston, D. Brown, S. J. Julier, and G. S. Schmidt. Military applications of augmented reality. In *NATO Human Factors and Medicine Panel Workshop on Virtual Media for Military Applications*, June 2006.
- [13] K. Mania, B. D. Adelstein, S. R. Ellis, and M. I. Hill. Perceptual sensitivity to head tracking latency in virtual environments with varying degrees of scene complexity. In *Proceedings of the First Symposium on Applied Perception in Graphics and Visualization (APGV'04)*, pages 39–47, Aug. 2004.
- [14] J. W. McCandless, S. R. Ellis, and B. D. Adelstein. Localization of a time-delayed, monocular virtual object superimposed on a real environment. *Presence: Teleoperators and Virtual Environments*, 9(1):15–24, 2000.
- [15] C. M. Robertson and B. MacIntyre. An evaluation of graphical context as a means for ameliorating the effects of registration error. In *IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 2007)*, pages 99–108, Nov. 2007.
- [16] C. M. Robertson, B. MacIntyre, and B. N. Walker. An evaluation of graphical context when the graphics are outside of the task area. In *IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 2008)*, Sept. 2008.
- [17] J. P. Rolland, W. Gibson, and D. Ariely. Towards quantifying depth and size perception in virtual environments. *Presence: Teleoperators and Virtual Environments*, 4(3):24–49, Winter 1995.
- [18] D. W. Sprague, B. A. Po, and K. S. Booth. The importance of accurate VR head registration on skilled motor performance. In *Graphics Interface*, pages 131–137, June 2006.
- [19] A. State, G. Hirota, D. T. Chen, W. F. Garrett, and M. A. Livingston. Superior augmented reality registration by integrating landmark tracking and magnetic tracking. In *SIGGRAPH 96 Conference Proceedings*, Annual Conference Series, pages 429–438. ACM SIGGRAPH, Addison Wesley, Aug. 1996.
- [20] B. Watson, N. Walker, P. Woytiuk, and W. Ribarsky. Maintaining usability during 3D placement despite delay. In *IEEE Virtual Reality*, pages 133–140, Mar. 2003.
- [21] M. Yeh, J. L. Merlo, C. D. Wickens, and D. L. Brandenburg. Head-up vs. head-down: Effects of precision on cue effectiveness and display signaling. In *Proceedings of the 45th Annual Meeting of the Human Factors and Ergonomics Society: Virtual Environments*, Oct. 2001.