An Invisible Head Marker Tracking System
for Indoor Mobile Augmented Reality

Chyigang Kuo\textsuperscript{a,*}, Taysheng Jeng\textsuperscript{b}, Itung Yang\textsuperscript{c}
\textsuperscript{a} Dept. of Architecture, Chaoyang Univ. of Technology, Taiwan.
\textsuperscript{b} Dept. of Architecture, National Cheng Kung Univ., Taiwan.
\textsuperscript{c} Dept. of Construction Engineering, National Taiwan Univ. of Science and Technology, Taiwan

*Corresponding author. Tel.: +886 933067607; fax: +886 4 23742339
E-mail addresses: chyigang@cyut.edu.tw (C. Kuo), tsjeng@mail.ncku.edu.tw (T. Jeng), ityang@mail.ntust.edu.tw (I. Yang)

Abstract

This research aims to develop a mobile augmented reality positioning system for indoor construction application by tracking the coordinates of the user and their angles of vision so as to realize three-dimensional indoor positioning with high environmental adaptability.

This research uses infrared invisible marker identification technique, and adopts the outside-in tracking mode to develop the Head Marker Tracking Augmented Reality (HMTAR) system, which can be mounted indoors or outdoors and has the extensive tracking capability, event memory mechanism, and scenario sharing mechanism. Evaluation results were obtained from Heuristic Evaluation and SUS evaluation. The results demonstrated a prospect for indoor augmented reality positioning technique.

Keywords: augmented reality, indoor positioning, mobile position tracking, invisible marker

1. Introduction

The ever-advancing Augmented Reality (AR) is gaining extensive applications in the construction field: such as, real-time 3D display of on-site construction progress [1], introduction of objects assembling procedures [2], 3D explanation of construction principle [3; 4], remote simulation of construction tools operation [5], design and revitalization in existing built environments [6] and many others. Among these cases, actual outdoor applications are mostly GPS-based tracking systems. However, the positioning errors and unreachability to indoor area of GPS signals have limited GPS-based AR application to outdoor fixed point control or merely for longer distance observation.
without providing close-range mobile operation. On the other hand, most indoor AR tracking systems are marker-based systems which have certain functional limitations too, namely, visual environmental impact due to the marker’s high black-and-white contrast, and marker-restricted environments (including constructions with no continuous flat surface, structures at historic site, unsuitable construction surface materials, water surface and mid-air prohibited from posting marker). It can be seen, therefore, that by solving the above-mentioned drawbacks of AR positioning systems, the overall AR application to the construction field can be increased to a greater extent. With this in mind, the study proposes a new mode for indoor construction application by summing up the advantages and disadvantages of the existing positioning techniques, and rethinking of trouble-shooting to those obstacles.

2. Analysis of AR position techniques for construction applications

Among all positioning techniques, Bhatnagar [7] has targeted four categories: mechanical, optical, ultrasonic and electro-magnetic in his rating of HMD (Head Mounted Display) positioning techniques. In addition to these four categories, other two categories: marker tracking [8, 9, 10] and natural feature tracking [11, 12, 13, 14] are added to form a more complete picture of positioning techniques and applied technologies, as shown in Table 1.

<table>
<thead>
<tr>
<th>Positioning Techniques</th>
<th>Applied Technology</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Mechanical Tracking</td>
<td>Robot arm, force feedback device;</td>
</tr>
<tr>
<td>2. Ultrasonic Tracking</td>
<td>Ultrasound waves (person positioning/spatial scanning);</td>
</tr>
<tr>
<td>3. Optical Tracking</td>
<td>Infrared rays, leaser, iGPs (person positioning/spatial scanning); target ball tracking, eyeball tracking.</td>
</tr>
<tr>
<td>4. Electromagnetic Tracking</td>
<td>GPS, DGPS, AGPS, TV-GPS, UWB, WAAS, RF, Bluetooth, ZigBee, RuBee</td>
</tr>
<tr>
<td>5. Marker Tracking</td>
<td>Marker recognition &amp; hidden marker recognition;</td>
</tr>
<tr>
<td>6. Natural Feature Tracking</td>
<td>PTAM (MonoSLAM), PTAMM, BazAR</td>
</tr>
</tbody>
</table>

Ratings of such positioning techniques have been seen in many previous related literatures. One prominent example is Gu and his colleagues, in “A Survey of Indoor Positioning Systems for Wireless Personal Networks,” [15] have proposed 8 criteria from 17 existing positioning devices both on market and under R&D: (1) Security and Privacy; (2) Cost; (3) Performance; (4) Robustness; (5) Complexity; (6) User Preferences; (7) Commercial availability; and (8) Limitations.
Focusing on the “accuracy” criterion, Institute of Geodesy and Photogrammetry, Swiss Federal Institute of Technology has integrated the detection range and relative accuracy of various positioning techniques [16]. As shown in Fig. A.1, appendix A, mechanical positioning has the highest accuracy, followed by optical tracking (including vision-based ProCam, Laser Tracker, and IGPS), and ultrasonic. Electro-magnetic (UWB, RFID, WLAN/Wi-Fi, AGPS and GSM) has the lowest accuracy.

In recent years, rapid development of natural feature tracking techniques, especially the most advanced marker-free PTAM [17], with its high mobility, greater tracking accuracy, free software library, and adaptability to indoor and outdoor applications, have basically met almost all the application requirements of AR systems. However, Lex van der Sluijs indicates that in large open spaces, where there is a large distance between the viewer and the nearest visual detail, the user has to move a large amount to even be able to begin visual tracking [18]. Also, in many buildings there are many self-similar locations. So if tracking is lost and has to be re-initialized, this is a problem if it has to happen based on visual clues alone [18]. This is especially true with initial indoor construction environment where only repeated similar structures and large space exist.

Meanwhile, re-scanning of feature points and reloading of virtual objects will be required to ensure displaying of AR images in the PTAM system with constant scenario change of the construction site when constructs, interior decorations or furniture are located in or removed from indoor construction environment. Such problems could possibly occur to indoor construction site with AR system of inside-out positioning technique, and probably this is where outside-in positioning techniques could be set in.

Outside-in positioning techniques include ultrasonic tracking [19], electro-magnetic tracking [20, 21], and optical tracking [22, 23, 24], with optical tracking having the highest positioning accuracy [16].

One of the most advanced optical tracking systems, Iotracker [23], is capable of conducting a 6-DOF (Degree of Freedom) positioning in a prepared environment. Nevertheless, the overly long path spanning between transmitting and receiving the IR rays subject the Iotracker to the interferences from surrounding thermal radiation and other IR rays, making the environmental light a very demanding condition. Meanwhile, although Iotracker allows simultaneously positioning by multiple users, the function is limited to the close similarity between the reflective target balls. To avoid mutual disturbance caused by the target balls, an IR camera is installed above each user for tracking purpose [25], provided that no user makes arbitrary moves into the field-of-view of other camera. This is the obstacle that Iotracker needs to overcome.

The above cases analysis leads us to conclude that the tracking technique for an AR system equipped in the indoor construction environment is suggested to have the following features: (i) outside-in tracking mode; (ii) 6DOF mobile tracking ability; (iii) adaptability to low illumination
environment; (iv) maneuverability at a variety of terrains; (v) posing less obstruction to construction progress; and (vi) lower impact on the positioning capability from surrounding construction changes.

3. System Establishment

To achieve the goals, the study proposes HMTAR (Head Marker Tracking Augmented Reality) system. HMTAR is an outside-in vision-based tracking system. By applying an opposite manner, its fully developed marker tracking techniques is thus transformed into a new position tracking system targeting the user’s headset. With the satellite cameras installed in the environment, the system can detect the coordinates and rotation direction of user’s head markers to perform real-time 6-DOF positioning tracking. The system can be maneuverably installed at various terrains, and the virtual objects can be positioned without placing any marker in the space.

HMTAR system comprises a position detection module (executed by AREyeSatellite program packet) and a mobile display module (executed by AREyeCenter package) (Fig.1 Left).

![Fig. 1. (left) Positioning Principle Framework. (right) Hardware Construct of Positioning Detection Module](image)

The position detection module consists of computer sets and USB IR webcams, no expensive hardware equipments are required. To conduct virtual object display calculation, first connect one set or several sets of satellite cameras (USB IR webcams) to a computer set. Each computer set then wirelessly transmits the detected positioning data to the server, which then transfers the integrated positioning data to user’s laptop computer (Fig.1 Right).

The mobile display module includes a positioning headset, and a transparent and ventilating functional backpack (Fig. 2). The headset is all black under naked eyes, but its invisible Eye Markers emerge under IR webcams (Fig. 3). The headset enables a multi-marker detection, to ensure at least one side-marker can be detected by the satellite camera when user travels, rotates, or bends or lifts
his/her head. Meanwhile, the headset uses a self-illuminating IR light source to reduce the IR rays travel distance and to enhance the system’s resistance to the disturbance from ambient lights (Fig. 4). Moreover, the IR positioning capability enables HMTAR to function in dark environments.

Fig. 2. Components of mobile display module. The user wears the i-Visor FX601HMD to watch the AR view.

Fig. 3. An all-black Eye Marker under naked eyes (left); Distinctive black-and-white positioning marker shown under IR camera (middle); Recognized markers and positioning data displayed on the screen (right).

Fig. 4. Structure and components of the headset: (left) Light metal body structure of the headset; (middle) Internal IR LED module; (right) EyeCamera attached to the headset brim to provide a first person AR view to the user (model: GoodTec My Cam, USB 2.0, maximum 8 mega pixels).
The core code of HMTAR is constructed on FLARToolKit, divided into seven modules, as shown in Fig. 5:

(i) FMS Module: Memorizing Satellite Camera’s own coordinate information under Flash Media Server, receiving user’s Eye Marker and auxiliary Target Marker coordinate information, and managing coordinate information generated by Center Module.

(ii) Center Module: Installed on user’s laptop for receiving Eye Marker positioning information transmitted by FMS, and computing 3D models generated by PV3D. Additional computing of Path Recoding Module and Video Recording Module are required when ARPlayer is ON.

(iii) FLARToolkit Module: The core code of the system installed on user’s laptop for calculating marker information detected by Satellite Camera and 6DOF movement coordinates by FlarDetector.

(iv) Satellite Module: Detecting Marker image, adjusting errors, integrating mean values, and transmitting the detected information to FMS Module.

(v) PV3D Module: Installed on user’s laptop for generating 3D objects and rendering materials according received coordinate information, and providing icon drawing functions.

(vi) Path Recording Module: When ARPlayer is ON, it can record user’s 6DOF path coordinate
information.

(vii) Video Recording Module: When ARPlayer is ON, it can record user’s viewed image information.

In case user’s motion exceeds the first satellite camera’s visual field, another satellite camera next in line can take over to form a 3D positioning array, where visible range and heights can expand. Preferably a zigzag array can be adopted to form a recognition area without any blind spots. Or alternately, a diametric array is adopted to gain the largest recognition range (Fig. 6).

![Fig. 6. Deployments of two satellite camera arrays](image)

After starting the system, the user holds the wireless mouse to click \[+\] / \[-\] buttons to add/remove virtual objects in the view of HMD (fig 7. left). Click “FollowMe” or “FollowMarker” to position selected object. Drag the handlebars on the Left to control motion drift, rotation and “Smooth” display of the virtual object.

First clicking FellowMe, followed by loading 3D virtual object, the loaded 3D object is then positioned in the center of user’s head. The 3D object will move away from the user until it reaches 1M of user’s front when the bar on the Right bottom is pushed to the far Right end (Fig. 7, Left). By shifting to Free, location of the virtual object is fixed.

Or alternatively, place a custom-made hexahedron - Target Marker (Fig.7. Right) - on the desired position of 3D object. Then check FollowMarker, 3D object will display on Target Marker. When shift to Free, 3D object will be positioned and will not disappear when Target Marker is later removed.
Fig. 7. (left) AR screen showing loading and setting of virtual object. (right) Target Marker.

HMTAR is capable of recording the whole process of AR dynamic frames, and memorizing the path, rotation angular parameters of user’s head movements (6-DOF parameters). Thus, all previous traveling trails can be cued by Map, Path, and Follow modes during later replay (Fig. 8).

A remote control agent device equipped with IP cam allows the user to generate AR frame by loading, deleting, or adjusting the 3D virtual objects with the operating mode exactly identical to the on-site operating mode (Fig. 9). Since the remote control of the agent device can be co-shared by multiply end-users, co-creation of the remote AR images is thus possible.
4. Evaluations and discussions

4.1 Positioning Accuracy

HMTAR’s core framework is based upon FLARToolKit. Under the same hardware specifications and environmental conditions, initial positioning accuracy of the system should equal to that of the ARToolKit. Related literature on ARToolKit positioning accuracy can be found in the related researches [26, 27], and ARToolKit official site [28]. Positioning accuracy of ARToolKit differs with factors such as icon sizes, recognition distance and angle etc. Although no resolution requirements are mentioned in the studies [26], it surely will be helpful that a camera equipped with high resolution, auto focus, auto whiteness balance, low illumination compensation, and back light compensation is adopted for long-distance recognition, dynamic recognition and environment with constant light changes. Therefore, rather than only performing well under a specific environmental condition with fixed positioning accuracy, HMTAR is created with adjustable feature to keep best accuracy under different lighting environment and operational conditions.

Firstly, HMTAR is capable of adjusting the initial positioning accuracy. The 6 DOF data and estimated distance (DIST) of Eye Marker (or Target Marker) are shown on top (and bottom) of the server screen. By shifting the “Zoom” value, the estimated marker size ratio could be calibrated; therefore the DIST value can be adjusted to match the real measured distance of Eye Marker or Target Marker (Fig.10).
Fig. 10. (left) The initial accuracy adjusting process of HMTAR; (middle) The 6 DOF data and estimated distance (DIST) of Eye Marker and Target Marker are shown on top and bottom of server screen; (right) The DIST value can be adjusted by dragging the “Zoom” bar in the server screen to match the real measured distance of Eye Marker or Target Marker.

However during the practical operation, HMTAR’s recognition performance, with its invisible marker recognition method, is unavoidably affected by the light wave decayed over distance. Other factors like IR Camera’s limited resolution, differences of IR Camera’s focus, different environmental lighting reflected by the black material of headset, and visual nebulization of aerial suspension particles, also cause instant-by-instant minor differences in the invisible marker’s recognition data. This in turn results in the jerking of virtual objects.

To overcome this phenomenon, secondly HMTAR has added a Smooth value, that is, a mean value is gained by calculating many coordinates generated by IR Camera and transmitted to PV3D for the virtual object’s coordinates. In other words, the larger the Smooth value adjustment, the larger amount of mean values can be fed to the system for calculation. So the smaller difference between mean values can lessen the jerking movement of the virtual object.

Furthermore, we have found that Smooth value is insufficient in stabilizing the jerking virtual object. This is because Smooth value is used as a positioning value by averaging many coordinate values. Even when the Smooth value is adjusted to its maximum, the jerking still exists, only reduced to a slower motion. So much so that a slow jerking can be witnessed even when the user stands still.

To tackle this obstacle, thirdly we have added a Bound Displacement Segment Valve, to allow a freehand to the user for UP/DOWN settings. During the initial positioning, the system will target the first coordinate parameter as a baseline value, and then using this baseline value to mark the UP/DOWN settings. In the following coordinate recognition process, only those exceeding the UP/DOWN settings among all coordinate values are selected as new parameters or the baseline value of the second coordinate. In this way, we can eliminate the minor recognition errors and the 3D virtual object will not make random jerking when user stands still. Besides, the system can keep up with the speed and increases the stability of the 3D virtual object by making more sensitive response when the user makes larger movement.
HMTAR is capable of reaching its optimal effect when the combined Bound and Smooth values are adjusted according to different environmental condition requirements. The codes of HMTAR about smooth and bound are listed in Appendix B.

To further enhance stability of 3D virtual object displaying on the AR screen, the following mechanisms are incorporated: Satellite camera perspective/focal length setup, satellite camera whiteness balance, brightness, contrast and other digital parameter controls, black headset internal IR LED brightness linear modulation.

The seemingly similarity between HMTAR and the improved marker tracking system, Intersense IS-1200 [29], are actually two entirely different operational concepts. Intersense IS-1200, belonging to the family of inside-out AR system, requires many contrasting black-and-white markers posting on the ceiling. Besides causing visual impact, it also fails to function normally at construction sites with no ceiling, complex roof structure and darker area or where marker is unsuitable, such as historic sites. Moreover, the posted markers will obstruct the construction process. All defects maintained above will relatively reduce the system’s adaptability in the indoor construction sites.

4.2 Evaluations

To verify HMTAR system’s usability, four AR experts (Chen, engineer at Human and Computer Interaction Center, National Industrial Technology Research Institute; Wang, engineer at Pitotech Co., Ltd.; Chang, engineer at Pitotech Co., Ltd.; and Huang, engineer at PolygonWorks Co., Ltd.) were invited to conduct Heuristic Evaluation [30, 31]. Three fixed satellite IR cameras are installed at the experiment site (Exhibition Hall, 7th floor, Design Institute Building of Chaoyang University of Technology) (Fig. 11 and Fig. 12).

Fig. 11. The setting locations of 3 satellite IR webcams in the exhibition Hall.
The four evaluators were guided to load a 3D object positioning at 1M ahead and then walk around to view it. The evaluators can freely adjust the 6 DOF of the 3D object, even remove and add the object into AR view again. After the operation, four evaluators were asked to make assessment based on 10 criteria of Heuristic Evaluation. Evaluation scores were divided into 5 scales: “Absolutely not,” 1 point, “Mostly not,” 2 points, “May be,” 3 points, “Mostly yes,” 4 points, “Absolutely yes,” 5 points. Final results are shown in the Table 2.

Table 2

Average scores of HMTAR on Heuristic Evaluation

<table>
<thead>
<tr>
<th>No.</th>
<th>Evaluation Criteria</th>
<th>Average Scores</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Visibility of system status.</td>
<td>4.75</td>
</tr>
<tr>
<td>2</td>
<td>Match between system and real world.</td>
<td>4.50</td>
</tr>
<tr>
<td>3</td>
<td>User control and freedom.</td>
<td>4.25</td>
</tr>
<tr>
<td>4</td>
<td>Consistency.</td>
<td>4.25</td>
</tr>
<tr>
<td>5</td>
<td>Error strategy.</td>
<td>3.75</td>
</tr>
<tr>
<td>6</td>
<td>Recognition rather than recall.</td>
<td>4.50</td>
</tr>
<tr>
<td>7</td>
<td>Flexibility and efficiency of use.</td>
<td>3.75</td>
</tr>
<tr>
<td>8</td>
<td>Aesthetics and minimalist design.</td>
<td>3.75</td>
</tr>
<tr>
<td>9</td>
<td>Help users recognize, diagnose, and recover from errors.</td>
<td>3.00</td>
</tr>
<tr>
<td>10</td>
<td>Help and documentation.</td>
<td>3.50</td>
</tr>
</tbody>
</table>
The assessment result indicates that the system has reached above average usability requirement. For construction application, HMTAR has ability to help designer present and position 3D virtual objects at site, so that workers can follow the 3D images to perform the construction task intuitively. However the lowest ratings in the evaluation fall on two items: “Help users recognize, diagnose, and recover from errors” and “Help and documentation,” indicating some auxiliary functions, such as “reset” or “help” functions, designed to help users operate the system remains to be strengthened.

Additionally, 40 students without AR system operating experience from Architecture Department and Interior Design Department were asked to simulate as visitors to the Design Gallery of CYUT to conduct SUS assessments for HMTAR (Fig. 13).

SUS (System Usability Scale) was developed in 1986 by British Digital Equipments Co., Ltd. to help business understand product’s ease of use, and as a comparison model to the last generation product or competitor’s product. It’s a frequently adopted subjective perception scale for product’s ease of use [33]. The SUS questionnaire, comprising 10 questions, is a cross-inquiring, 5-point Likert Scale, ranging from 1 (Strongly disagree) to 5 (Strongly agree) (Table 4). Calculation formula (1) [33] is as follows, wherein, S stands for item number. Total score ranges from 0 to 100, with the calculation value in positive correlation to the system’s usability.

\[
\{[(S1+S3+S5+S7+S9)-5] + [25-(S2+S4+S6+S8+S10)]\} *2.5 \quad (1)
\]

Fig. 13. Students are testing HMTAR.

For evaluating HMTAR, the students were asked to wear the system walking along a path back and forth, squat and stand up twice, and touch the pre-positioned virtual 3D objects around the gallery (Fig. 14). Each student only has 6 minutes to complete the actions, and then completes the SUS questionnaire. The purpose of the actions is to explore and simulate workers’ feeling while equipped with HMTAR to view and study the task at construction site. The evaluation results are shown in Table 4.
Fig. 14. The pre-positioned virtual 3D object in the AR view of HMTAR.

Table 4
SUS Evaluations of HMTAR

<table>
<thead>
<tr>
<th>Scores</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>What is your subjective feeling towards the system?</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Strongly disagree</td>
<td>Disagree</td>
<td>No opinion</td>
<td>Agree</td>
<td>Strongly agree</td>
</tr>
<tr>
<td>1. I think that I would like to use this system frequently.</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>31</td>
<td>5</td>
</tr>
<tr>
<td>2. I found the system unnecessarily complex.</td>
<td>1</td>
<td>28</td>
<td>7</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>3. I thought the system was easy to use.</td>
<td>0</td>
<td>1</td>
<td>4</td>
<td>26</td>
<td>9</td>
</tr>
<tr>
<td>4. I think that I would need the support of a technical person to be able to use this system.</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>21</td>
<td>18</td>
</tr>
<tr>
<td>5. I found the various functions in this system were well integrated.</td>
<td>0</td>
<td>3</td>
<td>10</td>
<td>19</td>
<td>8</td>
</tr>
<tr>
<td>6. I thought there was too much inconsistency in this system.</td>
<td>33</td>
<td>7</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7. I would imagine that most people would learn to use this system very quickly.</td>
<td>0</td>
<td>8</td>
<td>14</td>
<td>18</td>
<td>0</td>
</tr>
<tr>
<td>8. I found the system very cumbersome to use.</td>
<td>12</td>
<td>26</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>9. I felt very confident in using the system.</td>
<td>0</td>
<td>7</td>
<td>18</td>
<td>10</td>
<td>5</td>
</tr>
<tr>
<td>10. I needed to learn a lot of things before I could get going with this system.</td>
<td>4</td>
<td>30</td>
<td>6</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Number of person
Overall SUS evaluation reaches a value of 66.75, indicating a moderate to high level ease of use of HMTAR. According to the evaluation result, HMTAR might be helpful to inexperienced workers to understand the task process or final result in 3D on precise location, instead of reading task information from the construction drawing by themselves at the construction site.

5. Conclusions and Future Works

5.1 Conclusions

The study addresses the issue of which AR positioning technique is most adoptable for indoor construction application. Based on the outside-in tracking mode, this study has developed HMTAR system, to simultaneously recognize user’s coordinates and vision direction by employing IR invisible marker tracking technique. Since there is no need to place markers in the physical space, the high environmental adaptability feature allows the system to position virtual objects at the desired location, including mid-air, water surface, above flame, or even out field-of-vision, as suitable to ever-changing constructing sites. The headset of the system uses a self-illuminating IR light source to reduce the IR rays travel distance and to enhance the system’s resistance to the disturbance from ambient lights. Moreover, the IR positioning capability enables HMTAR to function in dark construction environments. Equipped with array extension positioning capability, the system can be installed (stationed and mobile) in indoor or outdoor continuous space.

A total of 6 mechanisms have been constructed in HMTAR to enhance positioning accuracy and image stability of 3D virtual objects, including 1 marker ratio calibration (Zoom value), 2 smooth value, 3 bound value, 4 satellite camera perspective/focal length setup, 5 satellite camera whiteness balance, brightness, contrast and other digital parameter controls, 6 black headset IR LED brightness linear modulation.

Heuristic Evaluation of HMTAR yields a mean value of 4 (most items score >3 points), demonstrating a high usability. S value resulted from evaluation of SUS is 66.75, indicating a moderate to high level ease of use perceived by the test subjects. The results of the two evaluations show that HMTAR could easily be operated by designers to position 3D virtual objects and by workers to explore and study construction task process and final results at construction site.

Although applying FLARToolkit library to construct the execution codes, HMTAR is created under outside-in tracking mode, which is totally different from all conventional inside-out tracking mode FLARToolkit applications whose posted markers will cause visual impact to the environment, obstruct the construction process, and fail to function normally at construction sites where marker is unsuitable.
5.2 Future Works

HMTAR is capable of positioning user’s eye coordinates and visual angles. Therefore, the system can calculate what objects and the coordinates of such objects entering the user’s visual frame. If the same coordinate system of the user is assigned to the real environment viewed by the user (by pre-establishing transparent 3D models of the environment “overlapping” the environmental images and ascribing with the same coordinates system) That is, the system can identify the “visual target” of the user. Given this, “Line-of-Sight Positioning Interactive Interface” is expected to be realized if hand gestures are combined with visual images of the user.

“Line-of-Sight Positioning Interactive Interface” is operated in a highly intuitive manner. For instance, in Fig. 10, the system will automatically turn on the light by the central control system, when user wearing “Line-of-Sight Positioning Interactive Interface” watches the light bulb and followed by dragging the bulb to the “ON” block on bottom Left, after finger circling the image in mid-air. The same applies to other mechanical facilities too. A simple hand gesture can activate the real facilities by linking to the central control system.

Possible realization of “Line-of-Sight Positioning Interactive Interface” depends on the realization of “Line-of-Sight Positioning” mechanism. Although still at its initial completion stage, the “Line-of-Sight Positioning” mechanism and “AR image recording, replay, and leading” capabilities of HMTAR, may be further introduced into the development and application of “Line-of-Sight Positioning Interactive Interface.” Possible future applications in construction projects include:

(i) Instant 3D display of future construction progress;
(ii) Flaws positioning for construction management;
(iii) Making real-time additional or alternative design on site;
(iv) Recording of AR images and replay or leading movement paths for worker instruction and education purpose.

References


[26] H. Kato, M. Billinghurst, Marker Tracking and HMD Calibration for a Video-based


Appendix A

Swiss Federal Institute of Technology has surveyed the detection range and relative accuracy of various positioning techniques [11].
Fig. A.1. Detection Range and Relative Accuracy of Various Positioning Applications.

Appendix B
The major codes of HMTAR related to Smooth and Bound design.

```
private function onChangeSmooth():void
    {_eyeManager._smooth = smooth.value;
     _targetManager._smooth = smooth.value;
     updateCenter();}
private function onChangeSmooth1():void
    {smooth.value = smooth.value - smooth.snapInterval;
     onChangeSmooth();}
private function onChangeSmooth2():void
    {smooth.value = smooth.value + smooth.snapInterval;
     onChangeSmooth();}
private function onChangeBound():void
    {_eyeManager._bound = bound.value;
     _targetManager._bound = bound.value;
     updateCenter();}
private function onChangeBound1():void
    {bound.value = bound.value - bound.snapInterval;
     onChangeBound();}
private function onChangeBound2():void
    {bound.value = bound.value + bound.snapInterval;
```
public function getEyeAVG():FMSEyeInfo
{
    if (_arrFMSEyeInfo.length == 0)
        return null;

    var m:int = _arrFMSEyeInfo.length;
    var found:Boolean = false;

    for( var i:int=0; i<m ;i++)
    {
        var obj:Object = _arrFMSEyeInfo.getItemAt(i);
        if( obj.marked )
        {
            found = true;
            //trace(_eye.n11)
            if(Math.abs(obj.n11 - _eye.n11) > _smooth/_bound)
                _eye.n11 = _eye.n11 - (_eye.n11/_smooth) + (obj.n11/_smooth);
            else _eye.n11 = _eye.n11;
            //trace(_eye.n11)
            if(Math.abs(obj.n12 - _eye.n12) > _smooth/_bound)
                _eye.n12 = _eye.n12 - (_eye.n12/_smooth) + (obj.n12/_smooth);
            else _eye.n12 = _eye.n12;
            if(Math.abs(obj.n13 - _eye.n13) > _smooth/_bound)
                _eye.n13 = _eye.n13 - (_eye.n13/_smooth) + (obj.n13/_smooth);
            else _eye.n13 = _eye.n13;
            if(Math.abs(obj.n14 - _eye.n14) > _smooth/_bound)
                _eye.n14 = _eye.n14 - (_eye.n14/_smooth) + (obj.n14/_smooth);
            else _eye.n14 = _eye.n14;
            if(Math.abs(obj.n21 - _eye.n21) > _smooth/_bound)
                _eye.n21 = _eye.n21 - (_eye.n21/_smooth) + (obj.n21/_smooth);
            else _eye.n21 = _eye.n21;
            if(Math.abs(obj.n22 - _eye.n22) > _smooth/_bound)
                _eye.n22 = _eye.n22 - (_eye.n22/_smooth) + (obj.n22/_smooth);
            else _eye.n22 = _eye.n22;
            if(Math.abs(obj.n23 - _eye.n23) > _smooth/_bound)
                _eye.n23 = _eye.n23 - (_eye.n23/_smooth) + (obj.n23/_smooth);
            else _eye.n23 = _eye.n23;
            if(Math.abs(obj.n24 - _eye.n24) > _smooth/_bound)
                _eye.n24 = _eye.n24 - (_eye.n24/_smooth) + (obj.n24/_smooth);
            else _eye.n24 = _eye.n24;
            if(Math.abs(obj.n31 - _eye.n31) > _smooth/_bound)
                _eye.n31 = _eye.n31 - (_eye.n31/_smooth) + (obj.n31/_smooth);
            else _eye.n31 = _eye.n31;
            if(Math.abs(obj.n32 - _eye.n32) > _smooth/_bound)
                _eye.n32 = _eye.n32 - (_eye.n32/_smooth) + (obj.n32/_smooth);
            else _eye.n32 = _eye.n32;
            if(Math.abs(obj.n33 - _eye.n33) > _smooth/_bound)
                _eye.n33 = _eye.n33 - (_eye.n33/_smooth) + (obj.n33/_smooth);
            else _eye.n33 = _eye.n33;
            if(Math.abs(obj.n34 - _eye.n34) > _smooth/_bound)
                _eye.n34 = _eye.n34 - (_eye.n34/_smooth) + (obj.n34/_smooth);
            else _eye.n34 = _eye.n34;
            if(Math.abs(obj.sclxyz - _eye.sclxyz) > _smooth/_bound)
                _eye.sclxyz = _eye.sclxyz - (_eye.sclxyz/_smooth) + (obj.sclxyz/_smooth);
            else _eye.sclxyz = _eye.sclxyz;
        }
    }

    // 192.168.0.100
/*trace(_eye.n11)
    _eye.n11 = _eye.n11 - (_eye.n11/_smooth) + (obj.n11/_smooth);
    trace(_eye.n11)
    _eye.n12 = _eye.n12 - (_eye.n12/_smooth) + (obj.n12/_smooth);
    _eye.n13 = _eye.n13 - (_eye.n13/_smooth) + (obj.n13/_smooth);
    _eye.n14 = _eye.n14 - (_eye.n14/_smooth) + (obj.n14/_smooth);
    _eye.n21 = _eye.n21 - (_eye.n21/_smooth) + (obj.n21/_smooth);
    _eye.n22 = _eye.n22 - (_eye.n22/_smooth) + (obj.n22/_smooth);
    _eye.n23 = _eye.n23 - (_eye.n23/_smooth) + (obj.n23/_smooth);
    _eye.n24 = _eye.n24 - (_eye.n24/_smooth) + (obj.n24/_smooth);
    _eye.n31 = _eye.n31 - (_eye.n31/_smooth) + (obj.n31/_smooth);
    _eye.n32 = _eye.n32 - (_eye.n32/_smooth) + (obj.n32/_smooth);
    _eye.n33 = _eye.n33 - (_eye.n33/_smooth) + (obj.n33/_smooth);
    _eye.n34 = _eye.n34 - (_eye.n34/_smooth) + (obj.n34/_smooth);
    _eye.sclxyz = _eye.sclxyz - (_eye.sclxyz/_smooth) + (obj.sclxyz/_smooth);
 */
 if( found )
    return _eye;
 else
    return null;
}