Automation in Construction 33 (2013) 37–47

Contents lists available at SciVerse ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autcon

Towards cloud Augmented Reality for construction application by BIM and SNS integration

Yi Jiao [a,c,d], Shaohua Zhang [c,d], Yongkui Li [b,*], Yinghui Wang [c,d], BaoMing Yang [e]

[a] School of Computer Science, Fudan University, 220 Handan Road, Shanghai, PR China
[b] School of Economics and Management, Tongji University,1239 Siping Road, Shanghai, PR China
[c] Development Center of Computer Software Technology, Shanghai, PR China
[d] Shanghai Ruanzhong Information Technology Co., Ltd. Shanghai, PR China
[e] Shanghai Lubansoft Co., Ltd. Shanghai, PR China

ARTICLE INFO

Article history:
Accepted 29 September 2012
Available online 29 October 2012

Keywords:
Augmented Reality
Cloud computing
BIM
Business social networking services
Integration
Web3D
Vision-based registration
Construction

ABSTRACT

Huge progress has been made on 'Augmented Reality' (AR) techniques such as registration, tracking, and display hardware. However, a construction AR system should be more convenient and combined with in-use applications to support multi-disciplinary users throughout construction lifecycle. This paper presents a video-based on-line AR environment and a pilot cloud framework. The contribution lies in two aspects: firstly, an environment utilizing web3D is demonstrated, in which on-site images are rendered to box nodes and registered with virtual objects through a three-step method; secondly, it is further extended to be "cloud" through federation of BIM (building information modeling) and BSNS (business social networking services). Technical solutions to key issues such as authoring, publishing, and composition are designed. The proposed environment is seamlessly integrated into in-use information systems and therefore enjoys greater usability. Implementations demonstrate how this framework and environment work. Although preliminary, it is conclusive for proof of concept.

1. Introduction

The concept of 'Augmented Reality' (AR) generally refers to superimposing computer generated virtual objects over real objects/scenes to produce a mixed world. Users can acquire additional information of real world by rendering this mixed overlay in devices such as head mounted displays, see through glasses, and hand held monitors. AR originates from 'virtual reality' (VR) and provides a semi-immersive environment; it emphasizes accurate alignment between real scenes and corresponding virtual world imagery in real time, or with right timing. As a widely acknowledged promising technology for enhancing human perception, it attracts many research attentions in a number of domains such as entertainment, tourism, and engineer [1–4].

The power of AR concept lies in the fact that it presents opportunities for users to interact with most interested digital content in a more natural way by virtue of effectual merging of the two environments [2]. A substantial of research work has been reported, which mainly falls into two dimensions: horizontal (application domain) and vertical (technology). [5–7] gave comprehensive reviews regarding AR implementation technologies, which identified tracking, registration and display as key issues to a successful AR system. Huge progress has been made on these issues; however, they are generally acknowledged as open questions. Horizontal research work indicates that different application areas require distinct implementation consideration, such as for fun and better experience in entertainment, for guidance in tourism, and for collaboration in engineering.

For a long time, AEC (architecture, engineering, and construction) sector has been regarded as a suitable and ripe application domain for AR. A number of fruitful findings have been reported [8–11], which demonstrates AR's potential and feasibility in this area. However, available literatures show that there is little matured AR industrial application in AEC practice. Most existing works confine themselves to research-stage lab prototypes [12,13]. The reasons are complex. To our observation, one important reason can be ascribed to inadequate consideration on usability. For example, many reported AR systems use special devices that are cumbersome, inconvenient, even causing ergonomics issues such as fatigue and discomfort [2,6]. Furthermore, most systems are designed as independent systems, isolated from dominating AEC applications such as BIM (building information modeling) and PMS (project management system). Without effective integration

* Corresponding author. Tel.: +86 21 65980907; fax: +86 21 65981368.
E-mail addresses: Jiaoyi@fudan.edu.cn (Y. Jiao), y.k.lee@126.com (Y. Li).

0926-5805/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.autcon.2012.09.018

mechanism, the performance of applying AR techniques to BIM/PMS is poor due to iteratively revised model versions in construction stages. In addition, AEC sector is highly fragmented, one-of-a-kind, and adversarial by itself. It has to deal with complex business relations, communications and process [14], which heavily requires multi-disciplinary collaboration among project participants and stakeholders. Current AR systems are mostly single-user oriented; although there are some researches investigating network and collaboration environment, they are not sufficient enough to fulfill AEC intrinsic requirements.

To achieve a prevalent industrial embracement in AEC sector, it is advisable to apply AR in mainstream AEC applications throughout construction lifecycle. We observe that in this applying process it is necessary to address, besides aforementioned key AR technique issues, at least three other critical issues: usability (easy and more convenient for use), virtual objects sourcing (automatic mapping with BIM), and collaboration (support multiple users and re-use, both synchronous and asynchronous). Innovative solutions and new research methods are required on these critical practical-oriented considerations.

In this paper, we propose a novel AR framework and environment toward this direction. We focus on these three issues by studying how AR technologies can be seamlessly integrated into mainstream AEC applications. As the integration scope is very wide, this study aims at a cloud AR application which supports on-line access and collaborative use. Our current research focus is to explore the feasibility of AR in novel cloud form, by integration with BIM and SNS (social networking services) applications. Initial outcome is a cloud AR formalization specifying its components and mutual relationship, and a preliminary implementation to get a proof of concept. Nevertheless, there would be a long way before a full-edged cloud AR application coming into truth, as some key AR techniques (such as registration) are still open questions. This research is part of a bigger effort to make construction life cycle more intelligent in an ongoing project called LubanWay, which, following our 10 years' building software development endeavor, is carried out from Jan. 2008 and consists of three academic and three industry partners in Shanghai, China.

2. Overview of related work and technology

In 1999, S. C-Y. Lu et al. [6] presented a comprehensive survey on Virtual and Augmented Reality (VAR) technologies for product realization, in which valuable road maps were made for AR research. Some important directions are listed in below: (1) from human point of view, a useful VAR system in practical applications must: be easy to use; accommodate a wide variety of human sizes; not cause fatigue or induce nausea; not require long periods of adaptation; (2) VAR technologies can affect each stage of product life-cycle, and bridge geographical gaps between collaborators by virtue of Internet, which is achieved by integration of VAR systems and product data management systems; (3) "at the center of all VAR systems is a digital model of the alternate world" that the systems try to represent. "Components of a complete VR system are all related to the creation, storage, manipulation, simulation and presentation of this digital model". Requirements to this model and its modeling environment are very high; and (4) a centralized CAD database is possible and preferable for VAR to be applied to the entire life cycle of AEC process.

These directions retain significant meaning today for developing AR systems, and in a way, well answered the question of why little matured AR system in AEC practice: the "digital model" and "project data management system" are quite immature in the past, which makes integration with AR an unrealistic task. However, new opportunities emerge with current technologies. In this research, a video-based, on-line AR application supporting multiple-user collaboration is proposed. This proposed application is seamlessly integrated with our proprietary as-built BIM (the "digital model") tools and 'business social networking services' (BSNS, the "project data management system") platform, storing and retrieving on-site images/videos in a unified logically centralized database. To the best of the authors' knowledge, there is no exact same research work as ours existing. For example, reported AR systems in [13,15–20] mainly focus on either tracking, registration, calibration, optical hardware, or other technical issues in specific construction field. In relation to one or several technical aspects, there are similar ones, some of which are examined in the following.

Technologies used in this framework mainly include: vision-based registration, web3D, BIM, SNS, and cloud computing. These technologies are at large not adopted by current AR systems.

To be more widely used in practice, it is critical for AR system to utilize virtual objects from in-use construction applications, e.g., BIM, the state-of-the-art construction data management technology, which originated from Computer Aided Design (CAD) and firstly deal with digital 3D graphic models. Although BIM is studied from various viewpoints [21–27], it is rarely examined as virtual object source for AR applications. Web3D technology, proposed by the open community of web3D Consortium, aims at promoting effective integration of 3D model with Internet/web and has now become an open ISO standard [28]. Developed in a broader context, web3D contains no specific features for AEC industry. However, some recent researches have noticed its great potential benefits. For example, in [29] the authors analyzed the combination possibilities and advantages of BIM and web3D and recommended the two communities to engage and collaborate for an integrated file format. This proposal is somewhat pioneering. However, the authors did not give further technical discussions and implementations, as we have done in the proposed framework. In [30], the authors proposed tangible interfaces for authoring 3D virtual scenes in AR environment. They used web3D 'Virtual Reality Modeling Language' (VRML) format to model and store 3D objects. Similar to our framework, they adopted vision-based AR and proposed an authoring and publishing solution too. However, they did not put their research under BIM environment. Neither did they give detailed technical elaboration. In [3], the authors proposed a vision-based AR registration method by selecting multiple planes in arbitrary positions and directions in real world. This method did not require a prior knowledge about multiple planes' geometrical relationship. The relationship was automatically estimated by computing a "Projective Space" of two selected reference images. That means, on construction site, users only need to place two markers freely and capture the real scene, then the augmentation can be accurately and automatically achieved. Users can watch virtual objects from favorite angles by moving camera. This method is elegant, easy for use, and can be extended to on-line applications by processing video stream at a frame rate of 10 fps, hence is adopted in our research. Again, the authors did not conduct their study under BIM background.

In another direction, collaboration requirements on AR system can be fulfilled by integration of real project management applications with defined access control/ permission mechanism. In [31], the authors proposed a collaborative system for product information visualization and augmentation. Their design rationale shares some similarity with ours. However, they limited their study in augmented annotation context, i.e., it is more an AR browser than a complete AR system. Moreover, no related access control mechanism is discussed. SNS and Cloud computing are new technologies well supporting collaboration. In [20], the authors proposed a SNS based mobile AR. In contrast with the introduction of access control mechanism in our proposed framework, they merely utilized the SNS website of twitter as a content finding database.

Our proposed framework and environment are based on solid practical foundations. Its characteristics mainly include: (1) web-based on line AR; (2) mobile AR; (3) video/vision-based AR; (4) AR integrated into as-built BIM applications; and (5) AR integrated into collaborative construction project management platform. As the foundation and indispensible contents of this integration research, we designed and developed a series of proprietary as-built cloud BIM tools, as well as a cloud BSNS collaboration platform for AEC.

3. Overview of proposed framework and system

3.1. Cloud AR for construction application

Cloud computing provides new opportunities for information system development in AEC sector. It refers to "a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., storage, applications) that can be rapidly provisioned and released with minimal management effort"[32]. Two cloud subjects concerning this research are: (1) Software as a Service (SaaS): a special-purpose software available through Internet which "does not require each end-user to manually download, install, configure, run the software applications on their own computing environments"; (2) Community cloud: a cloud shared by "several organizations and supports a specific community that has shared concerns" [32].

Applications in AEC sector are good candidates to be designed as cloud, especially as SaaS cloud and Community cloud. In this research, in order to study how AR technologies can be seamlessly integrated into existing AEC tools, we propose and develop a cloud BIM engine and a cloud BSNS application first. An AR framework and environment called Cloud AR for Construction Application are then proposed and integrated into these clouds. In general, AR systems fall into two categories: optical AR and video AR. The differentiation lies in the fact that in the former, users see real world scenes directly, while in the latter indirectly, usually through images/video streams taken by cameras. We choose to design a video AR, the reason is two-fold: 1) it can keep on-site real scenes recorded and stored permanently in pictures or videos, which can be used for further processing such as sharing with multiple users, auditing, and future facilities maintenance. These processing are valuable to AEC industry, which cannot be achieved by optical AR. 2) Video AR provides better registration and more convincing augmentation effects as both real and virtual objects are images/video streams [6]. It is more natural and easy for construction workers to perceive both objects in one mobile device's monitor. As registration errors are mainly caused by tracker system error and system lag, video AR can breakdown video streams into frames and have more time to synchronize these frames at system backend, hence better solve system lag error issue. Although this method is more dynamic and demanding in terms of continuous frame synthesis, some video blending techniques (such as chrome keying) can process the augmentation frame-by-frame and then render them in a reasonable flow rate (e.g., 10 frames per second) by using video data as tracking indicator. Therefore it supports on-line applications. Fig. 1 illustrates the proposed framework and system architecture.

3.2. Formalization

As an abstract generation of this research, a formal description of proposed Cloud AR for Construction Application is defined, which depict key elements and their mutual relationship. A cloud AR is a seven-tuple $\text{CARCA}\left(IR,VO,fa,AR,U,fp,fc \right)$, the semantics of which is defined as follows.

IR denotes the image set of real world scenes. There are video streams in practice. As a video can be decomposed into frames of image, it is treated as images too in this formal norm. *VO* is the set of virtual objects generated by an in-use BIM tool. *fa* is a mapping function between *IR* and *VO*. It corresponds to the abstraction of authoring function. *AR* represents the set of superimposed objects, which is the value domain of *fa*. In implementation it could be stored as either an independent entity, or merely some combing information of *VO* and *IR* elements. *U* is the user set denoting users having access right to superimposed objects. *fp* is the mapping function between *AR* and *U*, which depicts a 1: *N* relationship denoting the share of one *AR* element by many users. It corresponds to the abstraction of publish function. *fc* is the mapping function between *U* and *AR*, which depicts a 1: *N* relationship referring to simultaneous accessing of several *AR* objects by a higher rank user according to his business role. It corresponds to the abstraction of composition function. The working flow mechanism of proposed framework and system are presented in Algorithm 1.

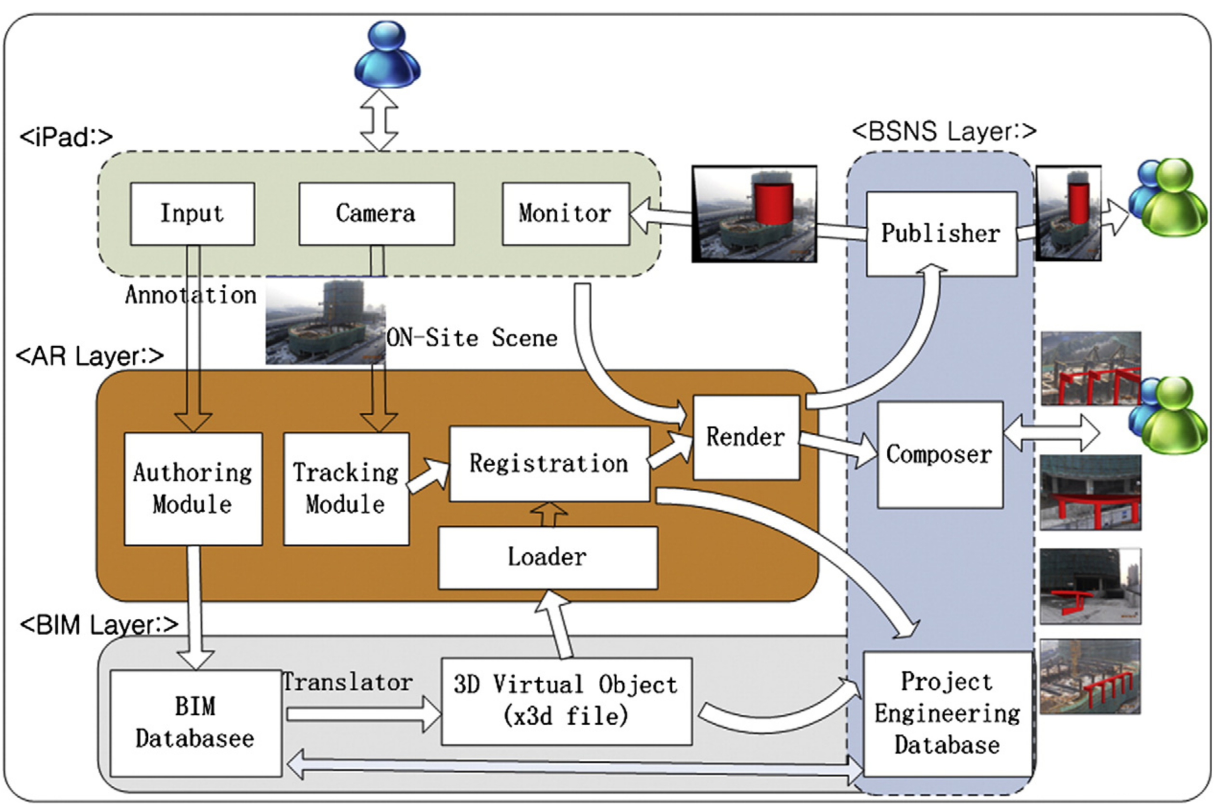

Fig. 1. Proposed framework and system architecture.

```
Algorithm 1
Input : UserVector, IR, IReferenceA, IReferenceB, AnnoVector, PEDB
      // IReferenceA, B : two reference picutures; AnnoVector : annotations made
      // by user, including user-id; PEDB : project engineering database
Output : ARstack
ImageStack = ARstack = nil
if (UserVector) // a compostion function, get all legall ARs of designated users in UserVector
    { ARstack = fc (UserVector) // fc : algorithm 3
      return }
else push IR into ImageStack // a normal AR function
while (ImageStack ) // for requring images, generating corresponding ARs
    {   pop         // get current image
        compute coordination according to IReferenceA, IReferenceB
        if (IR.fileformat == video)
            { decompose IR into frames
              push frames into ImageStack
              pop
            }       // treat video stream as frames
        BIMobject = Authoring (image, AnnoVector ) // retrieve corresponding viutual object in BIM
        transform BIMobject to web3D VO and load annotated information
        tile image to web3D box node, Registration(image.box, VO)
        store VO and registration data into PEDB as web3D x3d file
        AR = Render(x3d file)
        ViewList = fp(AnnoVector) // publish to multiple users, fp: algorithm 2
        push AR ⋃ ViewList into ARstack
    }
return ARstack
```

3.3. Framework foundation: in-use BIM cloud and BSNS cloud

This proposed framework consists of three interweaving parts: on-line AR, BIM cloud, and BSNS cloud (called MyLuban), which are integrated together so that AR can use real construction graphics from BIM and permission mechanism from BSNS. This section briefly introduces some BIM and BSNS knowledge regarding AR integration, such as 3D graphics generation and storage in BIM. For more details, we introduce them in another paper [33].

Our BSNS depicts the AEC sector's complex relationship in real world and act as a cloud project management data sharing platform for multi-disciplinary project teams and construction firms. It bears three features: (1) it has normal data collecting functions as well as SNS functions such as sharing information, establishing group, and discussing topic, through which users can communicate with strangers of common interests in other organizations; (2) a dynamic self-organized enterprise-sector-project-employee organizational structure, and a series of construction specific functions such as project monitoring, schedule control, and work breakdown structure (WBS), which enables collaborative performance of daily construct project works; and (3) a strict privacy and access control mechanism based on Role-Based Access Control (RBAC), which makes it suitable for business application instead of social activities.

As the designed AR virtual object source, three as-built BIM tools (civil engineering BIM, steel bar BIM, and installation BIM) are developed, which have the same client–server architecture and a unified storage mechanism. They are further encapsulated into web plug-ins to MyLuban and deployed as SaaS cloud application. HOOPS 3dGS™ (Graphics System) toolkit is adopted to generate 3D models; extensions are made to HOOPS through defining auxiliary classes (including attributes such as project fingerprint, level, position, and layer). Hence additional attributes are bound on 3D models and project engineering data are divided into two groups: geometric data (geometric models binding with extended attributes) and non-geometric data (other related engineering information such as bill, quota, component category, pricing mode, and floor). The former is grouped and stored in a HOOPS file by floor identifiers, i.e., all components in same floor form one file. The latter is stored in a relational database (e.g., SQL Server™, Oracle™) because it is more robust in depicting and storing large amount correlated information.

In order to effectively store HOOPS files, an open source NoSQL database of MongoDB is chosen. It is a distributed file system, most similar to relational database, and support cloud computing [34]. We further integrate relational database and MongoDB through designing a handler of GUID (Globally Unique Identifier) [35], which is generated by purpose-developed script engine when a HOOPS file is created. After generation, it is then automatically written as a HOOPS file attribute and registered into the relational database tuple in meantime.

Project management data gathered by MyLuban are stored in the same relational database. As embedded in MyLuban, data used in proposed AR environment are stored in these two databases. In order to get a logically centralized database, *UID* (username) are designed as foreign key to navigate between the two databases. Therefore, related AR overlays and BIM models can be automatically visited in project management activities through MyLuban, and vice versa.

4. AR on line

4.1. Basic implementation

The fundamental task of proposed framework is to design an on-line AR. To accomplish this, virtual objects need to be displayed and manipulated on web. Web3D is an excellent candidate in this respect. Its file format is VRML previously and now extensible 3D (x3d), with the latest specification of *v*3.3. The following part briefly introduces some knowledge concerning proposed framework. For more details about x3d, please refer to [28].

x3d in itself is a XML based file format, whose basic elements are *X3DNODE* and *X3DFIELD*. Nodes can contain specific fields with one or more children nodes which may, in turn, contain nodes. Hence forms a hierarchy of nodes (called scene graph), which is the basic unit of the x3d run-time environment. This hierarchy contains all the objects as well as their relationships, which are contained along several axes. Another hierarchy, called transformation hierarchy describes the spatial relationship of rendering objects. A x3d file may contain zero or more root nodes, which are not contained in other nodes. The descendants of a node are all of the nodes in its fields, as well as all of those nodes' descendants. The ancestors of a node are all of the nodes that have the node as a descendant. An illustrative example of a scene graph is given in below.

```
Transform { translatoin 1 2 3
            children [
                      Shape { geometry Box {}
                            }
                      Group { children [⋯]  }
                     ]
          }
```

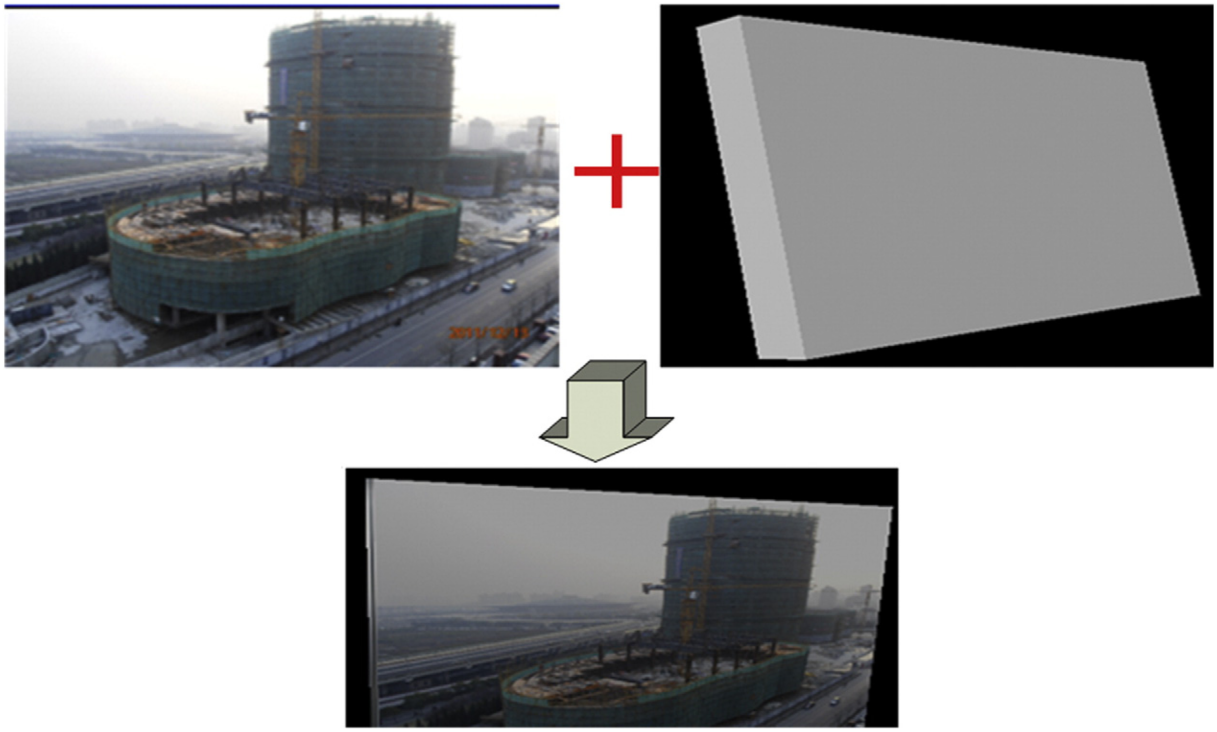

Fig. 2. Screenshot of rendering image to x3d box node.

With regards to AR, x3d specification provides components such as basic geometric node, CAD component, group nodes, and coordination transformation hierarchy.

As stated above, we adopt a video-based AR schema. The basic elements that the system manipulates are 2D images taken by a camera from construction sites and 3D virtual objects representing designed construction components in x3d format. Developing such an AR system means to find a way of accurately "*blending*" these two elements.

x3d does not directly support two dimensional images, although it provides some 2D nodes. Given this, the first step is to address this image compatibility issue. In the proposed framework, this issue is tackled by creating a cube (a "*box*" node in x3d) and tiling its surface with the image. First, a cube is created with three coordinates in the form of (x, y, z). By pre-defining an appropriate parameter in z axis (e.g., $z = 0.01$), the cube is approximately viewed as a plane with human vision perception. Second, using the "*url*" field in x3d *ImageTexture* node, an image defined in which can be rendered to *box* surface. Thus the manipulation of image is transferred to that of corresponding x3d *box* node. An illustration of this process is shown in below. Fig. 2 shows screenshots of this process.

```
< Transform translation = "-1.2 1 0
    <Shape>
        <Appearance>
            <ImageTexture url = " DSC_9482.jpg"/>
        </Appearance>
        <Box size = "4 3 0.01/>
    </Shape>
</Transform>
```

With this 2D–3D image transformation, next step is to accurately "*blend*" virtual construction components with transformed *box* surfacing input image. Technically, this is the open question of registration which would be detailed in the next section. We briefly present some preliminary knowledge about x3d coordination transformation here. Coordination in x3d is specified as a (x, y, z) triplet in a right-handed, rectangular coordinate system. The coordinate system in which the root nodes displayed is called the *world coordinate system*. Correspondingly, x3d defines *local coordinate systems* for children nodes in terms of transformations from ancestor coordinate systems. This is realized by the group node of *transform*. A transformation hierarchy includes all root nodes and their descendants that are considered to have particular locations in the virtual world. Part of the *transform* node specification is showed below.

```
Transform : X3DGroupingNode {
    addChildren          [X3DChildNode]
    removeChildren       [X3DChildNode]
    center               0 0 0    (-∞,+∞)
    children             []       [X3DChildNode]
    metadata             NULL     [X3DMetadataObject]
    rotation             0 0 1 0  [-1,1] or (-∞,+∞)
    scale                1 1 1    (-∞,+∞)
    scaleOrientation     0 0 1 0  [-1,1] or (-∞,+∞)
    translation          0 0 0    (-∞,+∞)
    bboxCenter           0 0 0    (-∞,+∞)
    bboxSize             -1 -1 -1 [0,+∞) or -1 -1 -1
}
```

4.2. Registration and tracking

In application integration scenarios, virtual objects to be displayed in AR system may consist of several construction components from in-use BIM engine. We choose to organize these components into one root node by x3d "grouping node", through defining offsets between world and local coordinate systems. Therefore, the registration issue is simplified to alignment of one image versus one virtual object.

Registration in this proposed framework is divided into three steps: (1) an automatic *initial* registration to align the image and virtual objects; (2) an automatic mapping of the first step in x3d; and (3) an additional manual *fine* registration to optimal alignment, if users feel needed.

The initial registration is a well studied open question; plenty of research findings have been reported on this issue [3,36,37]. In video-AR systems, this registration corresponds to computing virtual objects' 2D coordinates in background input image. Although called “initial”, it must provide a fairly accurate alignment effect. We choose a method proposed by [3] after a careful literature examination, in which multiple planes are identified from an input image to get better registration accuracy and simultaneous tracking stabilization. These planes can be in arbitrary positions and directions. For each plane i, an independent 3D coordinate system $(X_i - Y_i - Z_i)$ is designated. One plane is selected as a base plane with the special coordinate system of $(X_{base} - Y_{base} - Z_{base})$, in which the coordinates of the virtual objects are described. The input image is designated to a 2D coordinate system of $(x - y)$. The method introduces a virtual space $(P - Q - R)$, which is a 3D non-Euclidean coordinate system, defined by project reconstruction of two reference images, i.e., one same real world captured from two different viewpoints. The method works like follows: first, each plane i is projected to the virtual space by computing a transformation matrix T_i. The projection of the virtual objects to the virtual space could then be calculated by T_{base}. Second, transformation matrix P_i^{WP} is computed for each plane i, which projects $(X_i - Y_i - Z_i)$ to $(x - y)$ by making $Z_i = 0$ and then compute a homograph. The following equation holds:

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \simeq P_i^{WP} T_i^{-1} \begin{pmatrix} P \\ Q \\ R \\ 1 \end{pmatrix} \simeq Pi \begin{pmatrix} P \\ Q \\ R \\ 1 \end{pmatrix} \tag{1}$$

in which the coordinate $(P - Q - R)$ is projected into $(x - y)$ by each P_i. All P_is should coincide with each other and therefore could be integrated into one unique P. With P and T_{base}, the virtual objects could be projected into the image.

In our research, we select two reference images/planes by freely placing two barcode markers in real scene, and realize single frame image augmentation in which virtual object can be successfully overlaid to on-site picture. As would be discussed in Section 5.1, the barcode markers are also used for authoring virtual objects in BIM tool at meantime. For the registration accuracy and for more details about this initial registration, please refer to [3].

The multiple-plane method also uses mutual-reference of video frames as tracking mechanism and supports on-line video stream augmentation at a frame rate of 10 fps. In that case video stream are decomposed into frames, the planes appearing in different continuous frames correspond to tracking information. Currently we do not implement the video stream augmentation yet due to research schedule. As stated in Algorithm 1, we use a stack to store the decomposed frames and process the frames one by one. Replacement of a frame is further implemented by combination use of x3d Anchor node and in-line node and time sensor.

To apply this method in second step, two specific coordinate questions arise: (1) designation of virtual object coordination in real world (image coordination); and (2) projection of computed image coordination to x3d displaying coordination. The first question is an application specific issue. In AEC sector, an important subject is to compare as-built 3D status with as-designed one. In such cases, the coordination of as-designed objects to be designated in real world should be the same as the as-built component. However, the initial directions of virtual objects may still need to be adjusted manually; this can be done in fine registration step. For the second question, we designate the virtual object as a root node, and insert the *box* node (whose surface is the input image) as one of its children. According to the x3d transformation hierarchy, coordinates of virtual object are in *world coordinate system*. Suppose they are (X_1,Y_1,Z_1). The *box* coordinates in the same coordinate system are computed by the following formula: $(X_1 - x, Y_1 - y, \lambda\ Z_1)$, in which $X_1 - x, Y_1 - y$ prescribes the planar offset between image and virtual object; λ is a parameter which adjusts the offset of image and virtual object in Z coordination. These parameters are finally written into a *transform*

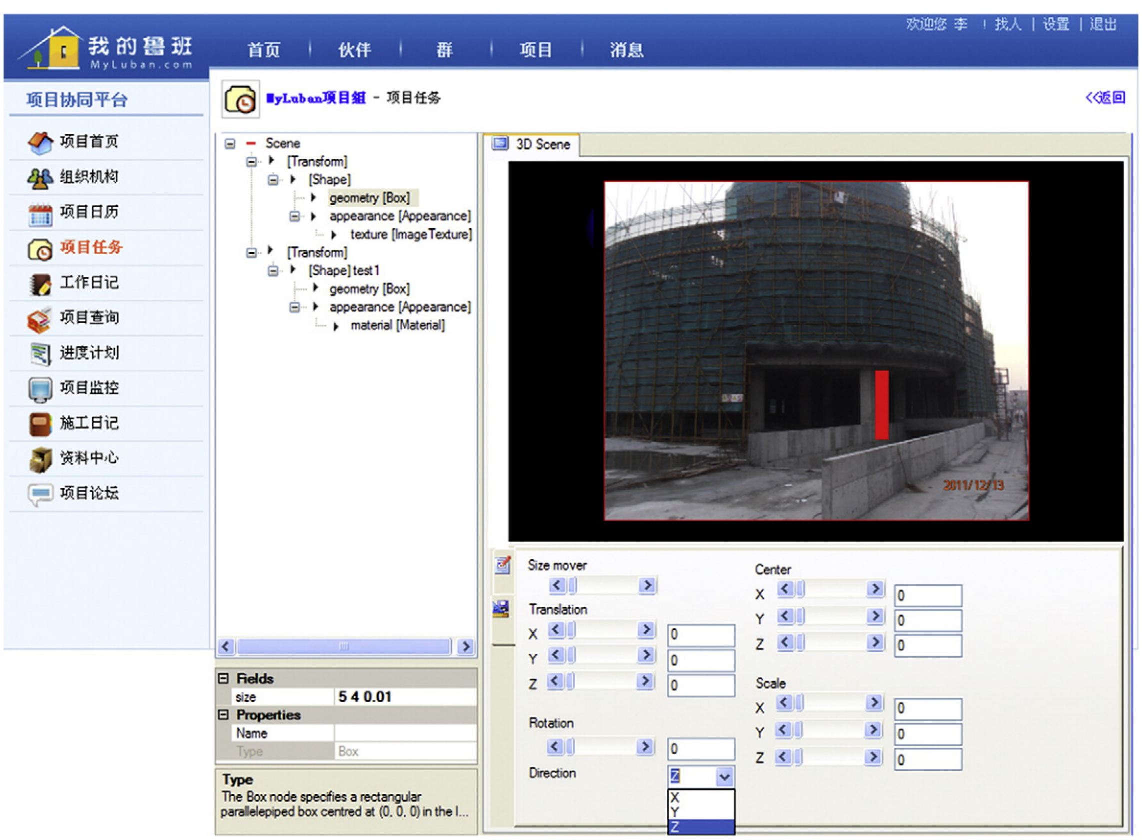

Fig. 3. Screenshot of fine registration.

node, i.e., ancestor of the *box* node. The default value of λ is set as 1/2, which means that the image is embedded in the middle of the virtual object. λ can be accurately manually adjusted by inputting a digital number in a textbox later. In this way, the occlusion issue is processed by using web3D's intrinsic mechanism: image is in fact tiled on the surface of *box* node, which is a 3D component; when $\lambda<1$, the virtual object to be augmented is "insert" into the 3D *box* node, which blots out the insertion part automatically.

After these two steps, an overlay mixture of input image and virtual objects can be automatically presented before users. For some cases, there may be further adjustments. Although some researchers argue that this should be avoided in AR system, in AEC sector this may be a special domain requirement because it can help users compare as-built and as-designed 3D status from multiple viewpoints and with a high accuracy demand. This manual adjust function provides opportunities for users to combine their personal expertise. The fine registration process can be performed in four dimensions: *X*, *Y*, *Z*, and a rotation scale. The former three adjusts the offset between image and virtual objects; the last one processes the augmentation as a whole by rotating it in different view angles. They are realized by a modification engine, in which users can manually adjust the (x, y, z) coordinates in the transform node which comprise the *box* node. A screenshot illustrating the fine registration process is shown in Fig. 3. System captures coordination of a virtual object and cascades them with below textboxes. Following their expertise and judgment, users can drag selected virtual object by mouse, or directly revise the three (x, y, z) coordination and rotation coordination in textboxes, until they feel satisfied with the alignment effect. The registration accuracy can reach a scale of 0.01° with digital number input.

5. AR for practice

5.1. Integrating AR and BIM

Integrating with as-built BIM tools and utilizing real construction data is one big advantage of our framework. In order to effectively integrate AR and BIM engine, we locate three issues to be addressed: (1) the virtual objects used in AR should come from those in BIM, either directly or indirectly; (2) given an on-site image, locate its corresponding virtual objects in BIM, i.e., authoring; and (3) the storage of the overlay information.

For the first issue, neither virtual object provided by existing BIM tools, including Autodesk, Bentley, and our HOOPS, can be combined with image directly. There must be some transform process between BIM virtual objects and AR ones. This transform process is accomplished automatically by a self-developed translator script from HOOPS to x3d.

A further question regarding this issue is BIM version management. BIM supports lifecycle collaboration among multi-disciplinary users. As a result, a BIM image may be revised iteratively and formed different versions. For the same virtual objects to be mixed in AR, there may be several storage locations denoting different BIM versions. Apparently, the correct version should be selected. The storage and management of BIM versions is a sophisticated issue. In our context, version is defined as a unique identifier to not only BIM geographical files/non-graphical records, but also related project management files/records. It is formally described as a Generalized List $UTs(g_c,g_p,g_{pr1},\ldots,g_{prn},T,B,N)$, where U denotes username (creator of this version), Ts denotes timestamp of version generation, g_c denotes GUID of this version, g_p denotes GUID of this version's parent version, g_{prl} to g_{prn} denotes reference versions, T denotes tags related to this version, B denotes branches related to this version, N denotes notifications from each user related to this version. We further define a construction data version container CDV, which is formally described as a Dynamic Generalized List series:

$$CDV\begin{cases} (\ ) \\ UTS \\ (UTS, CDV_1, CDV_2, \ldots CDV_n) \end{cases}.$$

Based on these definitions, a complete version of management mechanism including insert, update, and retrieval (traverse) algorithms is proposed and implemented. The default BIM versions to be mixed in AR are marked with a tag in the Generalized list. For detailed introduction of BIM version management, including how this mark process functions, please refer to our other paper [33].

For the second issue, we developed a special 2D barcode mechanism for authoring. It consists of two processes of encoding and decoding. The encoding function establishes a mapping between identifier of virtual objects and its location in real construction site. The identifier is designed in the form of (HOOPS file name, construction component *IDs*), in which HOOPS file name indicates specific floor number. The barcode is printed in common paper and placed at exact location of construction site. When using the proposed AR system, users take picture of this barcode with a mobile device and send it to application server. By receiving the barcode picture, a decoding module is triggered which scans and parses the image pattern to get parameters of HOOPS file name and component IDs. These parameters are sent to BIM database for inquiry. Target virtual objects can be located and rendered in BIM engine. Meantime, they are transformed into x3d file format.

For the third issue, the transformed x3d files and the images/information for augmentation are stored correlating with BIM and BSNS. As stated in Section 3.3, they form a logically centralized database. First, the image and x3d file to be mixed together must be stored in one file folder to avoid display error. Implementation of this design strategy is straightforward. For efficiency, a strategy is further defined to be in monthly interval, i.e., images and x3d files produced in one month are stored in the same folder. Second, we design a triplet (username, timestamp, GUID) to establish links between HOOPS file and x3d files. Username and timestamp are automatically designated as part of a x3d file name when a transformation occurs, meanwhile, GUID of corresponding HOOPS file is stored as a x3d file attribute. Finally, the x3d file name and attributes are stored in the relational database as BIM and BSNS do. Fig. 4 illustrates the entire storage mechanism, in which the fields of USERNAME, HOOPS id, CURRENT GUID, and PARENT GUID are used as foreign keys navigating database tables.

5.2. Publish and composition: integrating AR + BIM and BSNS

Publish and composition functions are another big advantage of proposed framework, which solve the collaboration issue in AR. Publish issue, in this research, refers to the exhibition of Augmented Reality overlay to multiple users at the right time. Composition, on the other hand, refers to a simultaneous exhibition of multiple Augmented Reality scenes produced by different users to one designated user. These functions are based on the integration of AR and BIM, and further integrated with our BSNS. They both need the support of BSNS's organizational structure and access control mechanism.

Traditional information systems cannot maintain multiple organizational structures in one application. To solve this problem, we design four categories of user accounts in BSNS cloud: individual account, project team account, sector account, and enterprise account. Each category account is an independent working unit and can be registered freely and independently. The establishment of organizational structure is realized by executing two atom operations: "join" and "removal". In backend, the changing organizational structure is processed by a Generalized List too [33]. Self organizing mechanism provides great probability and flexibility for multiple organizations working in one application. It provides necessary foundation for successful industry level cloud application. It creates and maintains organizational structure in a dynamic, bottom-up and decentralized manner. Each account is in charge of his scope independently.

Access control strategy in BSNS cloud consists of two parts: global and personalization. The basic global policy is designed in line with

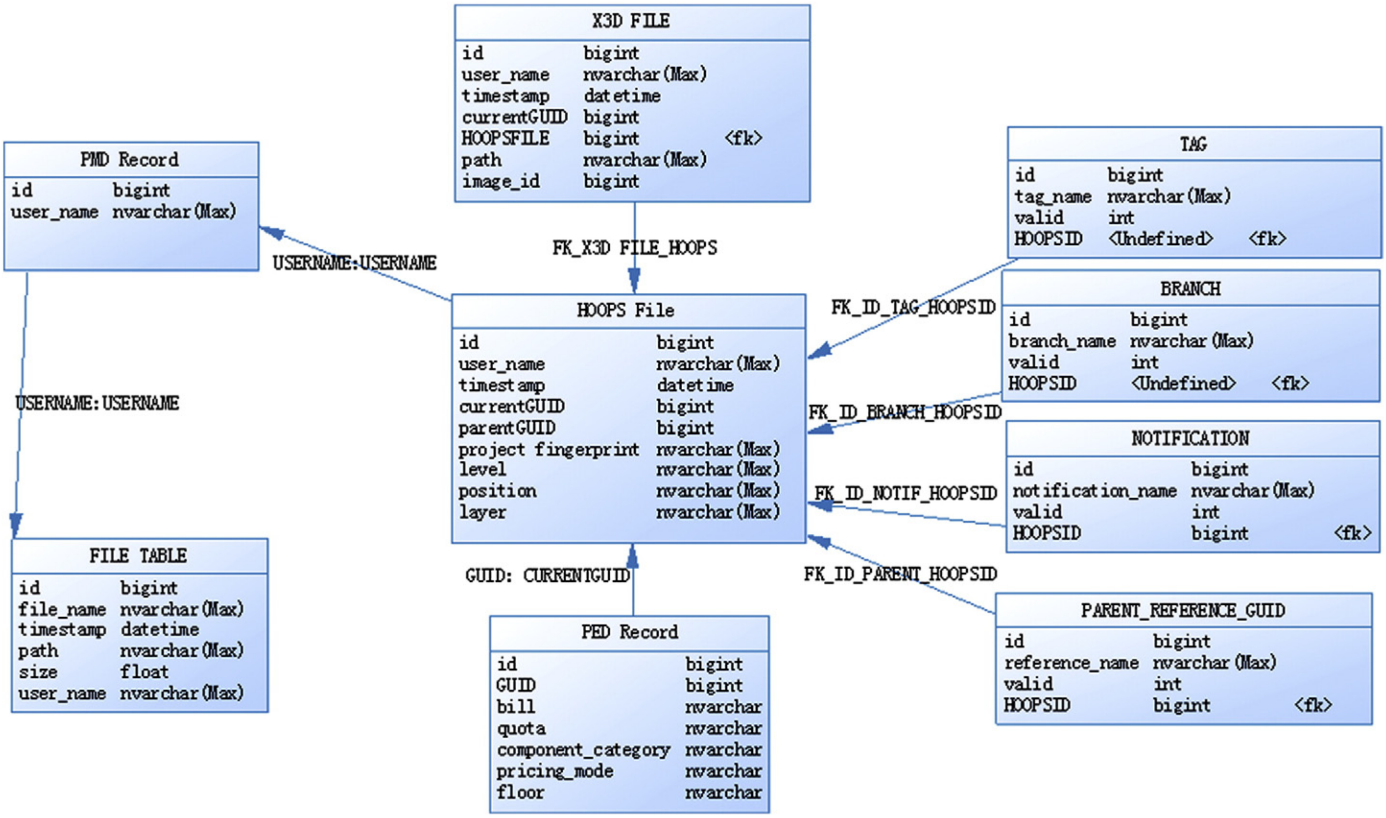

Fig. 4. Illustration of data storage in CARAC.

Role Based Access Control (RBAC) [38]. Personalization policy is designed according to user authorization. The mechanism is implemented in a decentralized manner; no single organization or user is responsible for the management of the overall access permissions [33].

These organizational structure and access control mechanism enable collaboration functions to be correctly performed in our cloud AR framework. The following Algorithm 2 and Algorithm 3 illustrate how publish and composition work, which correspond to *fp* and *fc* in formal description respectively.

Fig. 5. Screenshot of AR composition.

Algorithm 2 : *fp*
Input : *AnnotationVector*
Output : *UserList*

Get *CurrentUser* from *AnnotationVector*
Judge the role of *CurrentUser*, check *RoleList* from system backend to get *related roles*
using *CurrentUser* and *related roles* as index, traverse *Organizational Structure* at system backend
estalbish *RelatedUserList*
check *Permission List* from system backend, get *PermittedUserList*
$UserList = RelatedUserList \bigcup PermittedUserList$
return *UserList*

Algorithm 3 : *fc*
Input : *DesignatedUserList*
Output : *ARList*

check *PermittedUserList* from system backend
$EffectiveUserList = DesignatedUserList \bigcup PermittedUserList$
$ARList = \varnothing$
for *each user* $\in$ *EffectiveUserList*
{ query *database*, get related *GUID*
traverse BIM Generalized List CDV from system backend, retrieve corresponding x3d files
$ARList = ARList \bigcup (user, Retrieved\ x3d\ files)$ }
return *ARList*

Fig. 5 illustrates the screenshot of a composition function. A project manager wants to check the security and install work. With these two input usernames, four related augmentations stored in database are queried and presented; filenames are shown in the form of "username + timestamp".

6. Implementation and case study

We have developed an in-use BSNS application, three in-use BIM tools, and a preliminary AR online system, which jointly establish an integrated cloud environment. BIM engine is developed with C++ VS2005 and HOOPS-1800 3dGS™ toolkit. BSNS is developed with Java jdk 1.6, tomcat 6. AR system is developed with Java jdk 1.6, web3D x3d v3.3 and embedded into BSNS. A Single Sign On component is developed with Java jdk 1.6, which ensures users login once while accessing all the three systems. File translator from HOOPS to x3d is developed with C++ VS2005. Currently only limited scope of components is supported due to BIM graphics complexity. Databases are Oracle 11 g SE and MongoDB 1.8.0.

We conducted an Alpha-version test on this cloud AR implementation. The test was performed in a real construction project: Mansion of Shanghai Center; its general contractor, Shanghai Construction Group (SCG), subscribed our in-use cloud service of BIM and BSNS. A project account of SCP (Shanghai Center Project) was created and joined the enterprise SCG account in BSNS. Project manager, quantity surveyors, HVAC engineers, and water-supply-drainage engineers registered individual accounts, joined SCP account, and form a miniature hierarchical organization structure. Their access rights were designated accordingly. As a value-added service, AR functions were tried in on-site installation process. Before the test, initial installation models were built using cloud BIM tool. They were then revised 56 times and 72 installation model versions were generated, during which collision checks were performed to avoid model confliction. Finally a baseline model was agreed by all participants.

Test scenario was chosen on 16th floor, where two components of wind pipe and water-supply-drainage duct were selected from floor model. Floor version name and component IDs were encoded into barcode and printed on papers. On construction site, a quantity conveyor used an iPad2 logged in BSNS, took a picture of the barcode marker and uploaded it to the system. At server end, the system decoded this barcode image, queried database, found the target components, and transformed them to x3d models. The quantity conveyor then took pictures of the installation position (where the two barcode marker copies were placed at will serving as reference planes), uploaded them to the system, and designated publish to three persons: project manager, a HVAC engineer and a water-supply-drainage engineer. The system automatically tiled the picture to a x3d box node, registered it with transformed x3d components, and generated a new augmented x3d file version. These augmented information were stored in database automatically. During installation process, the HVAC and water-supply-drainage engineers logged in cloud BSNS, checked their task schedule, and got published mixed augmentation, which clearly indicated installation positions of different pipes. They turned to refer these AR augmentations by manually rotating them in various angles, with an accuracy of 0.01 scale degree, getting further knowledge about the right installation sequence, places and material size from BIM database. Each engineer repeated this augmented process after his installation by taking on-site pictures and barcode. In off-site office, quantity conveyor/project manager input the test floor number, which triggered composition function and displayed all the three augmented views in one window. By browsing and checking these views, he knew the pipe installation status, judged installation quality, and concluded that the whole on-site installation process was finished in line with design requirements and reached acceptance standard. In the testing moment, our system only supported the transformation of these two categories of simple component (exhaust duct/ air inlet pipe, service pipe/drainpipe) into x3d files; proportion and initial directions of virtual components were made manually after system registration. Even though, it demonstrates the three big advantages of our cloud AR environment compared with other approaches. First, it is rendered on web, accessed through Internet and merged with BIM and project management system, rather than an isolated desktop AR system. Therefore the virtual data used is from real construction project, as in the test case 3D pipe models, types, and sizes are from as-built BIM and BSNS. Second, it provides extended manual alignment function with an accuracy of 0.01 scale degree, which provides great flexibility for users combing their expertise in on-site construction performance. Third, the unique publish and composition functions allow one augmented model shared by different roles simultaneously according to their permission rights, as well as multiple augmented models rendered in mean time before one specific user for purposes such as audit, as in this test case project manager/quantity conveyor audited installation process jointly performed by HVAC and water-supply-drainage engineers. To the best knowledge of the authors, none of the three functions is reported in previous approaches. This test proves that preliminary as our cloud AR framework is, it can utilize as-built BIM virtual objects through internet access, support multi-disciplinary collaboration with different roles and access rights, and render augmentation in monitors of PC and mobile device such as iPad.

Each implementation technology has its strengths as well as its weaknesses; it is the specific application needs that affect different design choice [6]. We choose web3D technology because, compared with other standards such as Industrial Foundation Classes (IFC), it is naturally and intrinsically on web; it can be easily accessed by mobile devices and integrated with other in-use AEC applications, such as BIM and PMS. Further, x3d provides more consistent ways to integrate non-spatial "multi-media" data into graphics construction component, which can better support BIM. In contrast, IFC file is ASCII, large in volume, lacks multimedia data integration, and hardly to be effectively managed and rendered in real-time. Our current cloud AR application implementation is as yet preliminary because web3D technology is not AEC domain specific; a full translation from HOOPS graphic model to x3d scene graph needs more work. Nevertheless, the obstacles are lessening as web3D community is paying more attention to collaborate with AEC community. Newly establishment of web3D AR Working Group could exemplify this trend [28].

7. Conclusions and future work

AR has been introduced to a number of applications in AEC field with its potential value in areas such as progress monitoring, cost control, conflict avoidance, and tenant management. Much work has been conducted on AR techniques research and development. However, little effort has been laid on its broader application of integrating with other in-use AEC applications.

In this paper, we propose and develop an integrated cloud AR framework and environment, which focus on the key applying issues of usability, virtual object sourcing, and collaboration. We present our design rationale in a formal description and three algorithms. The proposed framework consists of a cloud BIM engine, a cloud BSNS application, and an on-line AR system. It utilizes web3D to render virtual objects, which originate from in-use BIM models. We adopt a multiple plane based automatic registration method, realize it in x3d context, and extend it by an optional fine registration step to improve accuracy and fulfill on-demand adjustment. Based on BSNS organizational structure and access control mechanism, we present publish and composition algorithms to ensure collaborative AR functions. Case study proves that this cloud AR can utilize real construction data from as-built BIM; by these algorithms an AR scenario can be designated to multidisciplinary users, and a user can monitor multiple AR scenarios created by different users. The proposed framework and system support special requirements (usability, sourcing virtual objects from BIMs, and collaboration) raised by broader application of AR technology in AEC domain; it is suitable and promising to be further extended and developed as daily tools in industrial applications.

Future work mainly includes: further implementation of video stream augmentation based on current image/singe-frame processing; improving our system to a larger scale of web3D embracement, such as more flexible graphics processing capacity, more robust translation ability from BIM models to x3d, and adopting new x3d specification. As web3D technology in itself is actively evolving, development of new features regarding AR application in AEC domain is on the way.

Acknowledgments

This work is partially funded by the Shanghai Rising-Star Program (B type, Grant no. 11QB1404300) and the National Natural Science Foundation of China (Grant no. 70902045). The authors would like to thank all industrial collaborators for their implementation of our research findings. We also thank Ruanzhong and Lubansoft for their agreement to publish real project data.

References

[1] X. Wang, P. Dunston, User perspectives on mixed reality tabletop visualization for face-to-face collaborative design review, Automation in Construction 17 (2008) 399–412.
[2] X. Wang, P. Dunston, Compatibility issues in Augmented Reality systems for AEC: an experimental prototype study, Automation in Construction 15 (2006) 314–326.
[3] Y. Uematsu, H. Saito, Multiple planes based registration using 3D projective space for Augmented Reality, Image and Vision Computing 27 (2009) 1484–1496.
[4] S.K. Ong, Y. Shen, A mixed reality environment for collaborative product design and development, CIRP Annals — Manufacturing Technology 58 (2009) 139–142.
[5] S. Kim, A.K. Dey, AR interfacing with prototype 3D applications based on user-centered interactivity, Computer-Aided Design 42 (2010) 373–386.
[6] S.C.-Y. Lu, M. Shpitalni, R. Gadh, Presented at opening session: virtual and augmented reality technologies for product realization, Annals of the CIR 48 (1999) 471–495.
[7] F. Zhou, H.B.-lirn Duh, M. Billinghurst, Trends in Augmented Reality tracking , interaction and display: a review of ten years of ISMAR, in: IEEE International Symposium on Mixed and Augmented Reality, IEEE Comput. Soc. Wash., DC, USA, September 15–18 2008, pp. 193–202, Cambridge, UK.
[8] D.H. Shin, P.S. Dunston, Identification of application areas for Augmented Reality in industrial construction based on technology suitability, Automation in Construction 17 (2008) 882–894.
[9] A. Hammad, J. Garrett, H. Karimi, Potential of mobile augmented reality for infrastructure field tasks, in: Proc. 7th International Conference on Applications of Advanced Technology in Transportation Engineering, ASCE, 2002, Am. Soc. of Civ. Eng., pp. 425–432, Cambridge, MA.
[10] K. Kensek, D. Noble, M. Schiler, A. Tripathi, Augmented Reality: an application for architecture, in: Proc.8th International conference on Computing in civil and Building Engineering, ASCE, 2000, Am. Soc. of Civ. Eng., pp. 294–301, Stanford, CA.
[11] J. Shen, Y. Wu, H. Liu, Urban planning using Augmented Reality, Journal of Urban Planning and Development 127 (3) (2001) 118–125.
[12] A.P. Gee, M. Webb, J. Escamilla-Ambrosio, W. Mayol-Cuevas, A. Calway, A topometric system for wide area augmented reality, Computers and Graphics 35 (2011) 854–868.
[13] D.H. Shin, P.S. Dunston, Technology development needs for advancing Augmented Reality-based inspection, Automation in Construction 19 (2010) 169–182.
[14] M. Ajam, M. Alshawi, T. Mezher, Augmented process model for e-tendering: towards integrating object models with document management systems, Automation in Construction 19 (2010) 762–778.
[15] J. Wither, Y.-T. Tsai, R. Azuma, Indirect augmented reality, Computers and Graphics 35 (2011) 810–822.
[16] F. Bosché, Plane-based registration of construction laser scans with 3D/4D building models, Advanced Engineering Informatics (2011), http://dx.doi.org/10.1016/j.aei.2011.08.009.
[17] T. Langlotz, C. Degendorfer, A. Mulloni, G. Schall, G. Reitmayr, D. Schmalstieg, Robust detection and tracking of annotations for outdoor augmented reality browsing, Computers and Graphics 35 (2011) 831–840.
[18] S. Lee, Ö. Akin, Augmented reality-based computational fieldwork support for equipment operations and maintenance, Automation in Construction 20 (2011) 338–352.
[19] D.H. Shin, W.-S. Jang, Utilization of ubiquitous computing for construction AR technology, Automation in Construction 18 (2009) 1063–1069.
[20] I. Song, I.-jae Kim, J.-in Hwang, S.C. Ahn, H.-gon Kim, H. Ko, in: Social Network Service based Mobile AR, Mus, 2010, pp. 175–178.
[21] N.Č. Babič, P. Podbreznik, D. Rebolj, Integrating resource production and construction using BIM, Automation in Construction 19 (2010) 539–543.
[22] W. Yan, C. Culp, R. Graf, Integrating BIM and gaming for real-time interactive architectural visualization, Automation in Construction 20 (2011) 446–458.
[23] J.P. Zhang, Z.Z. Hu, BIM- and 4D-based integrated solution of analysis and management for conflicts and structural safety problems during construction: principles and methodologies, Automation in Construction 20 (2011) 155–166.
[24] V. Singh, N. Gu, X. Wang, A theoretical framework of a BIM-based multi-disciplinary collaboration platform, Automation in Construction 20 (2011) 134–144.
[25] T. Cerovsek, A review and outlook for a 'Building Information Model' (BIM): a multi-standpoint framework for technological development, Advanced Engineering Informatics 25 (2011) 224–244.
[26] S. Fox, J. Hietanen, Inter-organizational use of building information models: potential for automational, informational and transformational effects, Construction Management and Economics 25 (2007) 289–296.
[27] R. Howard, B.-C. Björk, Building information modeling — experts' views on standardisation and industry deployment, Advanced Engineering Informatics 22 (2008) 271–280.
[28] web3D, Available from: http://www.web3d.org/realtime-3d/Accessed Dec.28, 2011.
[29] D.A. Campbell, M.A.M. Company, Building information modeling: the Web3D application for AEC, in: Web3D 2007 Symposium, ACM New York, NY, USA, April 15–18 2007, pp. 173–177, Umbria, Italy.
[30] J.Y. Lee, D.W. Seo, G.W. Rhee, Tangible authoring of 3D virtual scenes in dynamic augmented reality environment, Computers in Industry 62 (2011) 107–119.
[31] Y. Shen, S.K. Ong, A.Y.C. Nee, Augmented reality for collaborative product design and development, Design Studies 31 (2010) 118–145.
[32] S. Sakr, A. Liu, D.M. Batista, M. Alomari, A survey of large scale data management approaches in cloud environments, IEEE Communications Surveys and Tutorials 13 (2011) 311–336.
[33] Y. Jiao, Y. Wang, B. Yang, L. Li, L. Yuan, A cloud approach to unified Lifecycle data management in architecture, engineering, construction and facilities management: Integrating SNS and BIM, submitted to Advanced Engineering Informatics.
[34] MongoDB, . Available from: http://www.mongodb.org/Accessed Dec. 20, 2011.

[35] P. Leach, M. Mealling, R. Salz, A Universally Unique IDentifier (UUID) URN Namespace (IETF RFC 4122) (2005). Available from: http://www.ietf.org/rfc/%20rfc4122.txtAccessed Dec. 20, 2011.
[36] G. Simon, M. Berger, Reconstructing while registering: a novel approach for markerless augmented reality, in: Proceedings of the ISMAR, 2002, pp. 285–294.
[37] G. Simon, M.O. Berger, Real time registration known or recovered multi-planar structures: application to AR, in: Proceedings of the BMVC, 2002, pp. 567–576.
[38] ANSI INCITS 359–2004, American National Standard for Information Technology—Role Based Access Control, American National Standards Institute, Inc., NY, USA, 2004.

Yi Jiao is a Ph.D student at School of Computer Science, Fudan University. He is also an assistant researcher at Shanghai Development Center of Computer Software Technology. His research interests are cloud computing, semantic web, computer supported collaboration work, web service, augmented reality, and BIM.

Shaohua Zhang is an associate researcher at Shanghai Development Center of Computer Software Technology. He received his Ph.D. degree from School of Computer Science, Fudan University. His research interests are social network services, workflow, and cloud computing.

Yongkui Li is an associate professor at School of Economics and Management, Tongji University. He received his Ph.D degree from Tongji University. His research interests are complex construction project management, BIM, and Social Network in AEC.

Yinghui Wang is a researcher at Shanghai Development Center of Computer Software Technology. She received her Master degree from School of Computer Science, University of Shanghai for Science and Technology. Her research area is building information technology.

Baoming Yang is a senior expert at China Construction Industry Association. He received his Ph.D degree from Tongji University. His research interests are information technologies in construction (BIM and IT enabled collaboration).