ISO/IEC JTC 1/SC 24
Computer graphics, image processing and environmental data representation
Secretariat: BSI (United Kingdom)

Document type:	Text for CD ballot or comment
Title:	N 3943 ISO-IEC CD-18040
Status:	Please ensure your NSB votes by 2017-06-06 using the ISO balloting portal
Date of document:	2017-04-10
Expected action:	VOTE
Action due date:	2017-06-06
No. of pages:	43
Email of secretary:	charles.whitlock@bsigroup.com
Committee URL:	http://isotc.iso.org/livelink/livelink/open/jtc1sc24

Reference number of working document: **ISO/IEC JTC 1/SC 24 N 3943**

Date: 2017-03-15

Reference number of document: **ISO/IEC CD 18040**

Committee identification: ISO/IEC JTC 1/SC 24/WG 9

Secretariat: BSI

Information technology — Computer graphics, image processing and environmental data representation —

Live Actor and Entity Representation in Mixed and Augmented Reality

Document type: International Standard
Document subtype: subtype
Document stage: (20) Committee
Document language: E

STD Version 1

Contents

Page

Foreword ... 4

Introduction ... 5

1 Scope ... 7

2 Normative References ... 7

3 Terms, Definitions, and Abbreviated Terms ... 7
3.1 Terms and Definitions ... 7
3.2 Abbreviated Terms ... 11

4. Concepts of LAE representation in MAR ... 12
4.1 Overview ... 12
4.2 Components ... 14
4.2.1 LAE Capturer and Sensor ... 16
4.2.2 LAE Tracker ... 16
4.2.3 LAE Recognizer ... 16
4.2.4 LAE Spatial Mapper ... 16
4.2.5 LAE Event Mapper ... 16
4.2.6 Renderer ... 17
4.2.7 Display and User Interface ... 17
4.2.8 Scene Representation ... 17

5. LAE Capturer and Sensor ... 17
5.1 Overview ... 17
5.2 Computational View ... 17
5.2.1 LAE Capturer ... 17
5.2.2 LAE Sensor ... 18
5.3 Informational View ... 19

6. Tracker and Spatial Mapper for a LAE ... 20
6.1 Overview ... 20
6.2 Computational View ... 21
6.3 Informational View ... 22
6.4 An Example of LAE Tracking and Spatial Mapping in MAR ... 23

7. Recognizer and Event Mapper for a LAE ... 24
7.1 Overview ... 24
7.2 Recognizer ... 24
7.3 Event Mapper ... 25
7.4 Event Execution ... 26
7.5 Examples of LAE Recognizing and Event Mapping in MAR ... 27

8. Scene Representation for a LAE ... 28
8.1 Overview ... 28
8.2 Scene Description ... 29

9. Renderer ... 30
9.1 Overview ... 30
9.2 Computational View ... 30
9.3 Information View ... 30

10. Display and UI ... 31

11. Extensions to Virtual LAE ... 31

12. System Performance ... 32

13. Safety ... 32

14. Conformance ... 33

Annex A. Use Case Examples 34
A.1 3D Virtual Studio 34
A.2 Event Mapping of a LAE in a MAR world 34
A.3 Interactive Control of a Virtual Object in a MAR World by the Actions of a LAE 37
A.4 Augmenting a Real Object with Special Effects 38
A.5 3D Virtual Conference 39

Bibliography 40

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 18040 was prepared by Joint Technical Committee ISO/IEC JTC 1, Information technology, Subcommittee 24, Computer graphics, image processing and environmental data representation.

Introduction

This international standard defines the scope and key concepts of a representation model for a live actor and entity (LAE) to be included in a mixed and augmented reality (MAR) world. The relevant terms and their definitions, and a generalized system architecture, together serve as a reference model for MAR applications, components, systems, services, and specifications. It defines representing and rendering a LAE in a MAR world, and interaction interfaces between a LAE and objects in a MAR world. It defines a set of principles, concepts, and functionalities for a LAE applicable to the complete range of current and future MAR standards. This reference model establishes the set of required modules and their minimum functions, the associated information content, and the information models that shall be provided and/or be supported by a compliant MAR system. It includes (but is not limited to) the following content:

- Scope, which defines the extent which this standard intends to cover
- Normative references, which identify other standards documents indispensable to the application of this document
- Terms, definitions, and abbreviated terms that apply to including a LAE in a MAR world
- An introduction to the Mixed and Augmented Reality standards domain and concepts
- A representation model for including a LAE in a MAR world
- 3D modeling, rendering, and simulation of a LAE in a MAR world
- Attributes of a LAE in a MAR world
- Sensing representation of a LAE in a MAR world
- Representation of the interfaces for controlling a LAE in a MAR world
- Functionalities and base components for controlling a LAE in a MAR world
- Interactive interfaces between a LAE and a MAR world
- Interface with other MAR components
- Relationship to other standards
- Use cases

The objectives of this work are as follows:

- Provide a reference model for LAE representation-based MAR applications
- Manage and control a LAE with its properties in a MAR environment
- Integration of a LAE into a 2D and/or 3D virtual scene in a MAR world
- Interaction of a LAE with a 2D and/or 3D virtual scene in a MAR world
- Provide an exchange format necessary for transferring and storing data between LAE-based MAR applications

This International Standard has the following document structure:

In the prologue, the ***Foreword*** identifies the organization, and its nature, responsible for the establishment of the standard and the pertaining document publication. The ***Introduction*** gives a brief overview of the purpose and content of the document.

The main body of ISO/IEC 18040 is comprised of:

1. ***Scope*** defines the extent which this standard intends to cover.

2. ***Normative References*** identifies any normative references to related standards.

3. ***Terms, Definitions, and Abbreviated Terms*** lists and defines important terminologies and abbreviations for the respective domain of the standard.

4. ***Concepts of LAE Representation in MAR*** describe the concepts of LAE-based systems represented in MAR.

5. ***LAE Capturer and Sensor*** illustrates how a sensor captures a LAE in a physical world and a virtual world.

6. ***Tracker and Spatial Mapper for a LAE*** describe mechanisms to track the position of a LAE and specifies the role of a spatial mapper between physical space and the MAR space.

7. ***Recognizer and Events Mapper of a LAE*** describe mechanisms to recognize the behaviour of a LAE and specifies an association or event between a MAR event of a LAE and the condition specified by the MAR content creator.

8. ***Scene Representation for a LAE*** describes a scene, which consists of a virtual scene, sensing data, a spatial scene, events, targets, etc. for a LAE.

9. ***Renderer*** describes how the MAR world system renders the scene, LAE mapping, event, etc. for presentation output on a given display device.

10. ***Display and UI*** describes types of displays, including monitors, head mounted displays, projectors, haptic devices, and sound output devices for displaying a LAE in a MAR world.

11. ***Extension to Virtual Live Actor and Entity*** identifies and describes virtual LAE, such as virtual 3D model (avatar) and virtual LAE, such as real human model in a MAR system.

12. ***System Performance*** makes statements regarding any system performance related issues of a LAE in MAR.

13. ***Safety*** makes statements regarding any operational safety related issues of a LAE in MAR.

14. ***Conformance*** makes statements regarding any conformance related issues of a LAE in MAR.

ISO/IEC 18040 contains annexes:

A. ***Use Case Examples*** gives examples of representative LAE representation systems in MAR.

Bibliography lists all external references.

Information technology — Computer graphics, image processing and environmental data representation — Live Actor and Entity Representation in Mixed and Augmented Reality

1 Scope

This International Standard, Live Actor and Entity Representation in Mixed and Augmented Reality, defines a reference model and base components for representing and controlling a single LAE or multiple LAEs in a MAR world. It defines concepts, a reference model, system framework, functions, and how to integrate a 2D/3D virtual world and LAEs, and their interfaces, in order to provide MAR applications with interfaces of LAEs. It also defines an exchange format necessary for transferring and storing LAE-related data between LAE-based MAR applications.

This International Standard specifies the following functionalities:

a) Definitions for a LAE in MAR

b) Representation of a LAE

c) Representation of properties of a LAE

d) Sensing of a LAE in a physical world

e) Integration of a LAE into a 2D/3D virtual scene

f) Interaction between a LAE and objects in a 2D/3D virtual scene

g) Transmission of information related to a LAE in a MAR world

This International Standard defines a reference model for LAE representation-based MAR applications to represent and to exchange data related to LAEs in a 2D/3D virtual scene in a MAR world. It does not define specific physical interfaces necessary for manipulating LAEs. That is, it does not define how specific applications shall implement a specific LAE in a MAR world, but rather defines common functional interfaces for representing LAEs that can be used interchangeably between MAR applications.

2 Normative References

The following referenced documents are indispensable to the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 11072:1992, *Information technology — Computer graphics — Computer Graphics Reference Model*

ISO/IEC 18039, *Information technology — Computer graphics, image processing and environmental data representation — Mixed and Augmented Reality Reference Model, January, 2017*

3 Terms, Definitions, and Abbreviated Terms

3.1 Terms and Definitions

This standard uses a basic set of terms and definitions defined in ISO/IEC 18039, *Information technology -- Computer graphics and image processing – Mixed and Augmented Reality Reference Model,* and, additionally, the following terms and definitions apply.

3.1.1
Augmentation
Virtual object data (computer-generated, synthetic) added onto or associated with target physical object data (live video, a physical world image) in a MAR scene. Equally applies to physical object data added onto or associated with target virtual object data.

[ISO/IEC 18039:2016, definition 3.1]

3.1.2
Augmented object
An object with augmentation.

3.1.3
Augmented reality system
Type of mixed reality system in which virtual world data is embedded and/or registered with the representation of physical world data.

[ISO/IEC 18039:2016, definition 3.2]

3.1.4
Augmented virtuality system
Type of mixed reality system in which physical world data is embedded and/or registered with the representation of virtual world data.

[ISO/IEC 18039:2016, definition 3.3]

3.1.5
Display
Device by which rendering results are presented to a user. It can use various modalities, such as visual, auditory, haptics, olfactory, thermal, motion, etc. In addition, any actuator can be considered a display if it is controlled by a MAR system.

[ISO/IEC 18039:2016, definition 3.4]

3.1.6
Dynamic object
An object which can be translated, rotated, and scaled in a physical world or a virtual world.

3.1.7
Feature
Primitive geometric elements (e.g., points, lines, polygons, colour, texture, shape, etc.) and/or attributes of a given (usually physical) object used in its detection, recognition, and tracking.

[ISO/IEC 18039:2016, definition 3.5]

3.1.8
Geographic coordinate system
A coordinate system which provided by sensor devices for defining a location of LAE.

3.1.9
Head mounted display (HMD)
A device which displays stereo views of virtual reality, such as Samsung Gear VR, Oculus Rift, Google Cardboard, etc. It has two small displays with lenses and semi-transparent mirrors which can adapt to the left and right eyes.

3.1.10
Live actor and entity (LAE)
A representation of a living physical or real object, such as a human being, animal, or bird, in the MAR content or system. A live actor can be animated, moved, and interacted with virtual objects in a MAR world by capturing gesture from a camera. Entity refers to 3D objects and entities that exist in MAR content.

3.1.11
LAE recognizer
A MAR component that recognizes the output from a LAE capturer and a LAE sensor, then generates MAR events based on conditions indicated by the content creator.

3.1.12
LAE capturer
A MAR component that captures a LAE in a virtual world and a physical world, which includes depth cameras, general cameras, 360° cameras, etc. A LAE's video information will be processed by a LAE recognizer and LAE tracker to extract background or skeleton.

3.1.13
LAE sensor
A device that returns values related to a detected or measured condition or property related to a LAE. A LAE sensor may be an aggregate of LAE sensors.

[ISO/IEC 18039:2016, definition 3.20]

3.1.14
LAE tracker
A MAR component (hardware and software) that analyses signals from LAE capturers and sensors and provides some characteristics of a tracked LAE (e.g., position, orientation, amplitude, profile).

3.1.15
MAR event
An event which is triggered by the detection of a condition relevant to MAR content and augmentation (e.g. detection of a marker).

[ISO/IEC 18039:2016, definition 3.6]

3.1.16
MAR execution engine
A collection of hardware and software elements that produce the result of combining components that represent, on the one hand, the physical world and its objects, and on the other hand, those that are virtual, synthetic, and computer generated.

[ISO/IEC 18039:2016, definition 3.7]

3.1.17
MAR experience
The human visualization of and interaction with a MAR scene.

[ISO/IEC 18039:2016, definition 3.8]

3.1.18
MAR scene
The observable spatiotemporal organization of physical and virtual objects. It is the result of a MAR scene representation being interpreted by a MAR execution engine. A MAR scene has at least one physical object and one virtual object.

[ISO/IEC 18039:2016, definition 3.9]

3.1.19
MAR scene representation
A data structure that arranges the logical and spatial representation of a graphical scene, including the physical and virtual objects that are used by the MAR execution engine to produce a MAR scene.

[ISO/IEC 18039:2016, definition 3.10]

3.1.20
Mixed and augmented reality system
Term synonymous with *mixed reality system*[1].

[ISO/IEC 18039:2016, definition 3.11]

3.1.21
Mixed reality continuum
Spectrum spanning physical and virtual realities according to a proportional composition of physical and virtual data representations (originally proposed by Milgram et al. [1])

[ISO/IEC 18039:2016, definition 3.12]

3.1.22
Mixed reality system
A system that uses a mixture of representations of physical world data and virtual world data as its presentation medium.

[ISO/IEC 18039:2016, definition 3.13]

3.1.23
Marker
In the context of a MAR system, a marker consists of metadata embedded in a MAR background that specifies the location of a superimposed object.

[ISO/IEC 18039:2016, definition 3.27]

3.1.24
Movable volume
A volume in which a LAE is movable in a physical world or in a virtual world.

3.1.25
Natural feature
A feature that is not artificially inserted for the purpose of easy detection/recognition/tracking.

[ISO/IEC 18039:2016, definition 3.14]

3.1.26
Physical object
A physical object that is designated for augmentation with virtual data representation.

[ISO/IEC 18039:2016, definition 3.15]

3.1.27
Physical reality
Term synonymous with the physical world itself or a medium that represents the physical world (e.g., live video or a raw image of the physical world).

[ISO/IEC 18039:2016, definition 3.16]

3.1.28
Physical world
Spatial organization of multiple physical objects.

[ISO/IEC 18039:2016, definition 3.17]

[1] The word “augmented” is often used together with the word “mixed”.

3.1.29
Point of interest
A single target location or a collection of target locations. Aside from location data, a point of interest is usually associated with metadata, such as an identifier and other location specific information.

[ISO/IEC 18039:2016, definition 3.18]

3.1.30
Physical coordinate system
A coordinate system that enables locating a LAE and is controlled by a geospatial coordinate system sensing device.

3.1.31
Spatial registration
The establishment of the spatial relationship or mapping between two models, typically between a virtual object and a target physical object.

[ISO/IEC 18039:2016, definition 3.21]

3.1.32
Static object
An object which cannot be translated, rotated, and scaled in a physical world or a virtual world.

3.1.33
Target image
A target object represented by a 2D image.

[ISO/IEC 18039:2016, definition 3.22]

3.1.34
Target object
A target physical object designed or selected to allow detection, recognition, and tracking (and, finally, augmentation).

[ISO/IEC 18039:2016, definition 3.23]

3.1.35
Virtual live actor and entity
A virtual reality representation of a LAE. The virtual LAE is obtained by a 3D capturing technique and can be reconstructed, transmitted, and compressed in the MAR world. A virtual LAE can be captured in one place and transmitted to another place in real time using holography technology.

3.1.36
Virtual object
A computer-generated entity that is designated for augmentation in association with a physical object data representation. In the context of MAR, it usually has perceptual (e.g., visual, aural) characteristics and, optionally, dynamic reactive behaviour.

[ISO/IEC 18039:2016, definition 3.25]

3.1.37
Virtual world or environment
Spatial organization of multiple virtual objects, potentially including global behaviour.

[ISO/IEC 18039:2016, definition 3.26]

3.1.38
World coordinate system
A universal system in computer graphics that allows model coordinate systems to interact with each other.

3.2 Abbreviated Terms

This standard uses a basic set of abbreviated terms defined in ISO/IEC 18039, *Information technology -- Computer graphics and image processing – Mixed and Augmented Reality Reference Model,* and, additionally, the following abbreviated terms apply.

3.2.1
LAE
Live actor and entity

3.2.2
MAR
Mixed and augmented reality

[ISO/IEC 18039:2016]

3.2.3
LAE-MAR
Live Actor and Entity representation in Mixed and Augmented Reality

3.2.4
S-camera
Stereo camera

3.2.5
D-camera
Depth camera

3.2.6
RGB
Red Green Blue

3.2.7
SID
Sensor identifier

3.2.8
EID
Event identifier

3.2.9
VR
Virtual reality

3.2.10
HMD
Head mounted display

3.2.11
UTM
Universal Transverse Mercator

3.2.12
FOV
Field of view

3.2.13
GNSS
Global Navigation Satellite System

4. Concepts of LAE representation in MAR

4.1 Overview

As illustrated in ISO/IEC 18039: **Mixed and Augmented Reality (MAR) reference model**, MAR represents a continuum that encompasses all domains or systems that use a combination of reality (e.g. live video) and virtuality representations (e.g. computer graphic objects or scene) as its main presentation medium [1,2]. Figure 1 illustrates the MAR that defined according to mixture of reality and virtuality representations. The real environment refers to the physical world environment where a LAE and objects are located. Augmented reality refers to the view of real world environment whose elements included LAE and objects can be augmented by computer-generated sensory. Augmented virtuality is the virtual environment that physical world elements included LAE can be mapped and interacted within. Virtual environment commonly refers to virtual reality that is the computer-generated realistic images and hypothetical world that replicate a real environment. In Figure 1, a LAE wears a HMD device to see the virtual world and interacts directly with virtual objects.

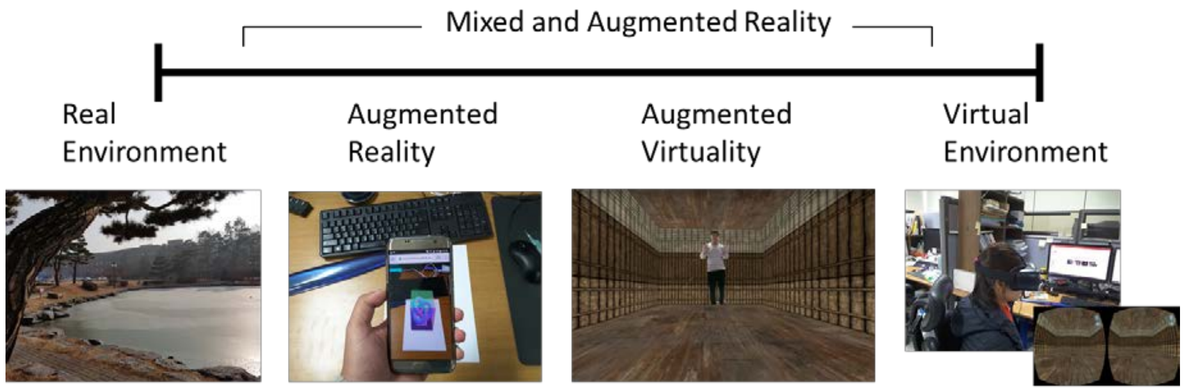


Figure 1. Mixed and Augmented Reality (MAR)

This clause describes the concepts of LAE representation in a MAR world based on the MAR reference model (MAR-RM) of ISO/IEC 18039, which includes objectives, embedding, interaction, and functions of the system for representing a LAE in a MAR world. In general, an actor is an individual who portrays a character in a performance. In our case, an actor is a human captured by a depth camera or a general camera that can then perform actions that will be embedded into a MAR world. A 3D object that exists in a MAR world and that can interact with a live actor is called an entity. The entity can be moved or interact with an actor's motion via an event mapper. A LAE in this standard is defined as a representation of a physical living actor and an object in a MAR content or system. For example, human beings, birds, and animals are all represented as LAEs in a MAR world.

Figure 2 shows the examples of LAE representation in a MAR world which consist of 2D virtual world and 3D virtual world that can be described as the following.

2D virtual world + LAE (a) – spatial mapper

Figure 2(a) shows a LAE integrated into a 2D virtual world that is a real or virtual image. The LAE can be captured from general camera and/or depth camera sensors [3]. This figure shows a real-like action where a man is captured by cameras in a green screen studio and is integrated as an actor into a 2D virtual world image of the White House.

3D virtual world + LAE (b) – spatial mapper

Figure 2(b) shows multiple LAEs integrated into a 3D virtual world. This scenario can be applicable in various situations, such as news studios, education services, virtual surgical operations or games [4-6]. It supports an integrative combination application of 3D videoconferencing, reality-like communication features, presentation/application sharing, and 3D model display within a mixed environment.

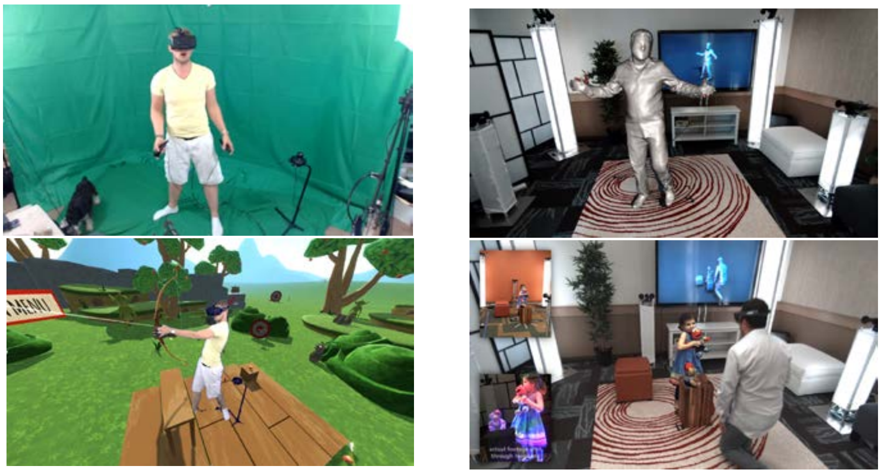
3D virtual world + LAE (c) – spatial mapper + event mapper

Figure 2(c) shows a MAR world constructed by integrating a 3D virtual world and a live actor [7]. The live actor interacts with objects in the 3D virtual world by using a joystick or by motion captured by a depth camera. An HMD device is used to display 360^{o} 3D views, including real time and real-like action, in the virtual world [8]. The figure shows a man in the studio wearing a HMD through which he sees the bow sling training field, and handling a joystick. By handling the joystick, it appears that he is handling the arrow and bow sling. As a result, he can shoot at objects in the virtual world.

Figure 2(d) shows a virtual actor and entity with representation in the physical world [9]. The virtual LAE and LAE can communicate and interact with each other, for example to have a natural face-to-face conversation. When combined with MAR, this technology allows a LAE to see, hear, and interact with a virtual LAE in a 3D virtual space, just like a real presentation in physical space.

(a) A LAE integrated into a 2D video virtual world after Chroma-keying [3]

(b) Two LAEs integrated into a 3D virtual world



(c) A LAE interacting with a virtual object in a 3D virtual world [7]

(d) Virtual representation of a LAE in a MAR world [9]

Figure 2. Examples of LAE representation in a MAR world

Once a LAE in the physical world is integrated into a 3D virtual world, its location, movements, and interactions should be represented precisely in the 3D virtual world. In a MAR application, a LAE that needs to be embedded in a 3D virtual world must be defined, and then information, such as the LAE's location, actions, and sensing data from a handled device, must be able to be transferred between the physical world and the 3D virtual world, and between MAR applications. This work is intended to define how a LAE can be managed and controlled with its properties in a 3D virtual world; how a LAE can be embedded in a 3D virtual world; how a LAE can interact with virtual cameras, virtual objects, and AR content in a 3D virtual world; and how MAR application data can be exchanged in heterogeneous computing environments.

4.2 Components

A LAE in a MAR world can be captured from the physical world, then represented in a 3D virtual world, and can interact with cameras, objects, and AR content in the 3D virtual world according to an input of sensing information.

In order to provide a 3D virtual world with the capability of representing a LAE based on the MAR-RM, the MAR system requires the following functions:

▪ Sensing of a LAE in a physical world from input devices such as a (depth) camera

▪ Sensing of information for interaction from input sensors

▪ Recognizing and tracking a LAE in a physical world

- Recognizing and tracking events made by LAEs in a physical world
- Recognizing and tracking events captured by sensors in a physical world
- Representation of the physical properties of a LAE in a 3D virtual world
- Spatial control of a LAE in a 3D virtual world
- An event interface between a LAE and a 3D virtual world
- Composite rendering of a LAE into a 3D virtual world

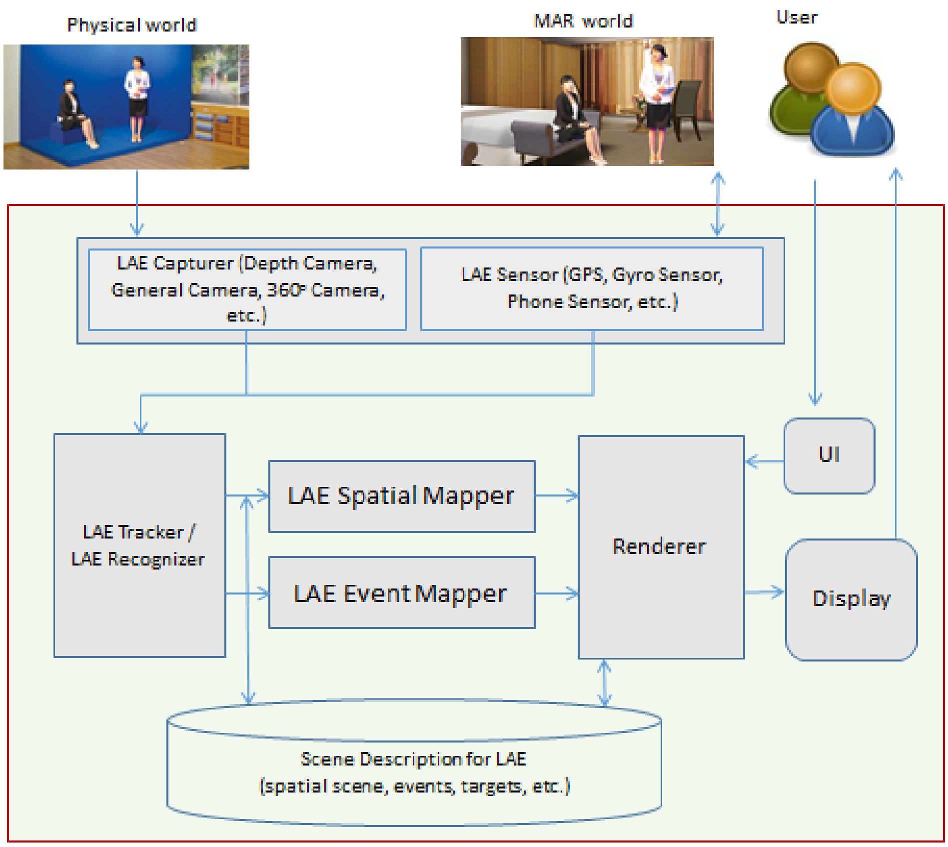


Figure 3. The system framework for a LAE in a MAR world

Conceptually, the system for implementing a MAR world with LAEs includes five components necessary for processing the representations and interactions of the LAEs to be integrated into the 3D virtual world. The configuration of the MAR system is shown in Figure 3.

Table 1. Attributes of a LAE in each component

Dimension	Types	
	Input	**Output**
LAE capturer/sensor	Physical world signal	Sensor data related to representation of a LAE
LAE recognizer	Raw or processed signals representing the LAE (provided by sensors) and target object specification data (reference target to be recognized)	At least one event acknowledging the recognition
LAE tracker	Sensing data related to the representation of a LAE	Instantaneous values of the characteristics (pose, orientation, volume, etc.) of the recognized target signals

Dimension	Types	
	Input	**Output**
LAE Spatial mapper	Sensor identifier and sensed spatial information	Calibrated spatial information for a LAE in a given MAR scene
LAE Event mapper	MAR event identifier and event information	Translated event identifier for a LAE in a given MAR scene
Renderer	MAR scene data	Synchronized rendering output (e.g., visual frame, stereo sound signals, motor commands, etc.)
Display/UI	Render scene data/user actions	Display output/response of user actions

4.2.1 LAE Capturer and Sensor

A MAR world with a LAE can receive a captured image, or a sequence of captured images, and sensing data triggered by the LAE in the physical world. The data for the LAE representation can be obtained from sensing devices/interfaces and is classified into two types: one for the LAE, and one for the sensing data for the LAE's actions. The data for the LAE will be used to perform both spatial mapping and event mapping of the LAE into the 3D virtual space, and the sensing data will be used to perform event mapping between the LAE and the 3D virtual environment.

The MAR system can use devices such as Web cameras and depth-like devices. For Web cameras, the sequence of images will be captured and the images themselves will be used as input. A depth camera is a movement sensing interface device for capturing the movement of a LAE. The device consists of an RGB camera, 3D depth sensors, multi-array microphones, and a motorized tilt. The device uses an IR (infrared radiation) camera and an IR laser projector to take 3D depth information.

4.2.2 LAE Tracker

Modules are required to preprocess the captured image, or a sequence of captured images, and the sensing data obtained from the sensing devices and interfaces. Preprocessing examples include Chroma-keying, color conversion, and background removal. Tracking a LAE refers to finding the location of the LAE in each image of the sequence. The modules can be implemented by OpenCV (*Open Source Computer Vision Library*) [10], which is a library of programming functions mainly aimed at real-time computer vision.

4.2.3 LAE Recognizer

Recognition refers to finding and identifying the actions of a LAE in the captured image, or a sequence of captured images, and the sensing data. When a sensor processes the output of a MAR component and generates MAR events, the LAE recognizer identifies each event and matches the event with an event ID from the database. The LAE recognizer analyzes signals of LAE motion or interaction from the physical world by comparing them with a local or remote signal. Then, an event function will be processed.

4.2.4 LAE Spatial Mapper

A LAE in a physical world is embedded into a 3D virtual world in a MAR application. The spatial mapper's role is to support more natural movement of the LAE within the 3D virtual world. The LAE spatial mapper provides spatial information, such as position, orientation, and scale, between the physical world space and the MAR world space by applying transformations for calibration. The LAE spatial mapper maps the physical space and the LAE into the MAR scene by supplying explicit mapping information. The mapping information can be modelled by characterizing the translation process of a sensor's given information.

4.2.5 LAE Event Mapper

A LAE in a physical world will act by aiming. These actions can be reflected in the 3D virtual world where a LAE is participating and interacting with virtual cameras, virtual objects, and AR content. The LAE event mapper's role is to support the actions of the LAE. It creates an association between a MAR event and the events which are identified or recognized by the LAE recognizer.

4.2.6 Renderer

The results of spatial and event mappers for a LAE are transferred into the rendering module. This module integrates the results and a 3D virtual scene, and displays the final rendering result of the integrated outcomes. The MAR scene can be specified by various capabilities of the renderer, thus the scene can be adapted and simulation performance can be optimized.

4.2.7 Display and User Interface

The final rendering result of the integrated outcomes of LAEs and a MAR world can be displayed on a variety of devices, such as monitors, head mounted displays, projectors, scent diffusers, haptic devices, and sound speakers. A user interface provides users with a way to modify the state of the MAR scene. A WebVR [11,12] is proposed to display a rendered scene on a HMD device [13].

4.2.8 Scene Representation

The MAR scene describes all information related to LAEs in the MAR environment. This information consists of sensing data, spatial scene, events, targets, etc. The MAR scene observes the spatiality of physical and virtual objects, and has at least one physical object and one virtual object.

5. LAE Capturer and Sensor

5.1 Overview

A LAE in a physical world can be captured from hardware and (optionally) software sensors that are able to measure any kind of physical property. As referenced in ISO/IEC 18039, two types of sensors, “capturer” and “sensor”, are used to embed the LAE into the virtual world and to perform an interaction between the LAE and objects in the virtual world. The most common “capturer” sensors are video and/or depth cameras which capture a physical world as a (depth) video containing live actors and entities. The video is used to extract the LAE and its actions to be processed by the recognizer and tracker components. The actions, especially, may affect interaction between the LAE in the physical world and objects in the virtual world. The target physical object can generate physical or nonphysical data which can be captured from a “sensor”. The (non) physical data can be used to detect, recognize, and track the target physical object to be augmented, and to process the interaction. The sensing data is input into the recognizer and tracker components.

5.2 Computational View

A LAE capturer/sensor module defines the functionalities of components and their interfaces for sensing a LAE. It specifies the services and protocols that each component exposes to the environment. This module provides two types of sensors, “capturer” and “sensor”, for sensing a LAE.

5.2.1 LAE Capturer

Various types of cameras, including general cameras, depth cameras, and 360^{o} cameras, can be used to capture a LAE. Figure 4 shows a LAE capturer that captures the physical world, including a LAE, as a video, depth image, and skeleton which is used in a MAR scene. A LAE can be extracted in a pre-processing step for the LAE tracker and/or recognizer by using video processing methods such as background removal, filtering, and Chroma-keying. An extracted LAE can not only be embedded into a virtual world, but it can also be used to identify a MAR event. That is, a LAE can be used as input to the LAE tracker and/or recognizer.

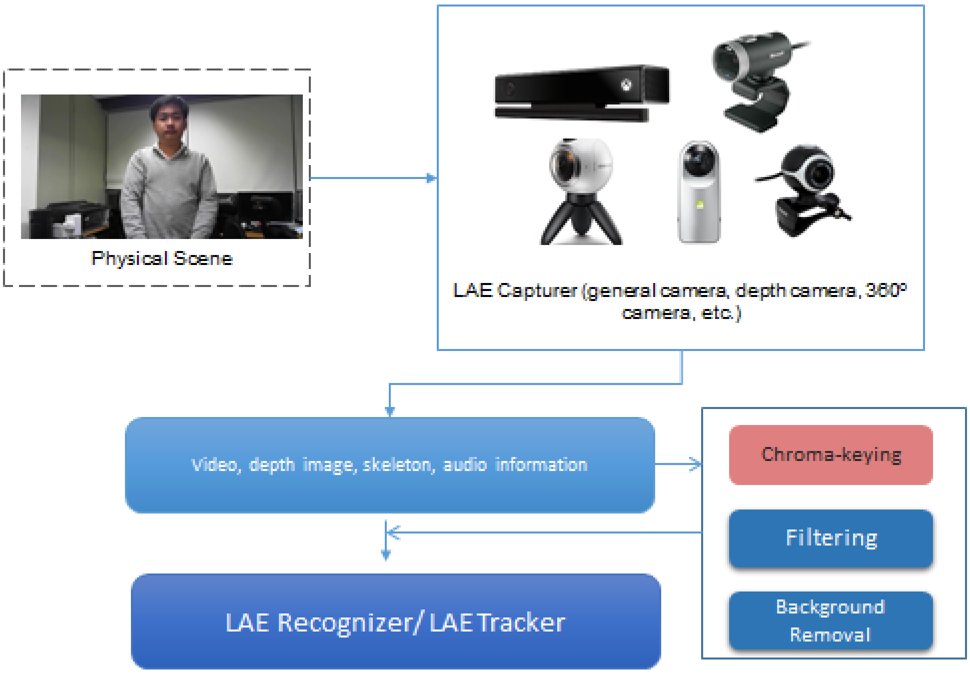


Figure 4. LAE capturer

5.2.2 LAE Sensor

A LAE sensor can measure various physical properties and interpret and convert the observations into digital signals related to the LAE. Figure 5 shows sensor capturing actions related to LAE activities. It shows that the captured data can only be used to compute the context in the LAE tracker and LAE recognizer, or it can be used to both compute the context and contribute to the composition of the scene, depending on the nature of the physical property or type of sensor device [14]. There are many types of sensors that can be used to control virtual objects, virtual cameras, and augmented objects by a LAE in a MAR environment. These sensors can generate different results depending on their properties, such as position, direction, geographic coordinate system, time, motion, etc. Especially, the output of sensors can be filtered and regenerated as high quality data.

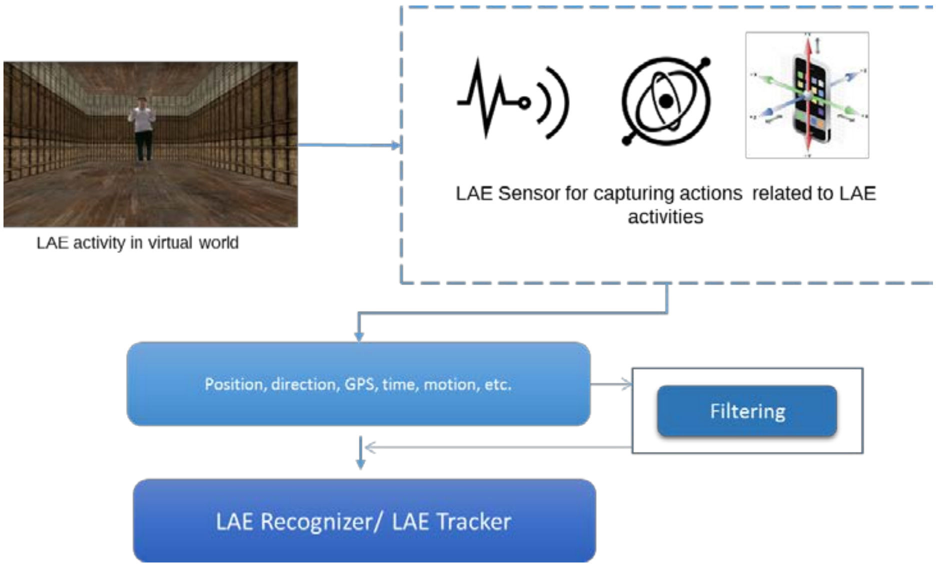


Figure 5. Sensor capturing actions related to LAE activities

As an example, Figure 6 shows how a smart phone sensor can be used to generate a gyroscope sensor, accelerometer, digital compass, ambient light sensor, proximity sensor, barometer, Hall Effect sensor, magnetometer, pedometer, and so on. A gyroscope sensor is used to generate the angular velocity of the rotational angle per unit of time to detect the rotation of the phone. An accelerometer sensor is used to measure acceleration forces and dynamics to sense movement of the phone or vibration. A digital compass sensor is used to detect the geographic coordinate system, direction, and navigation information of the phone.

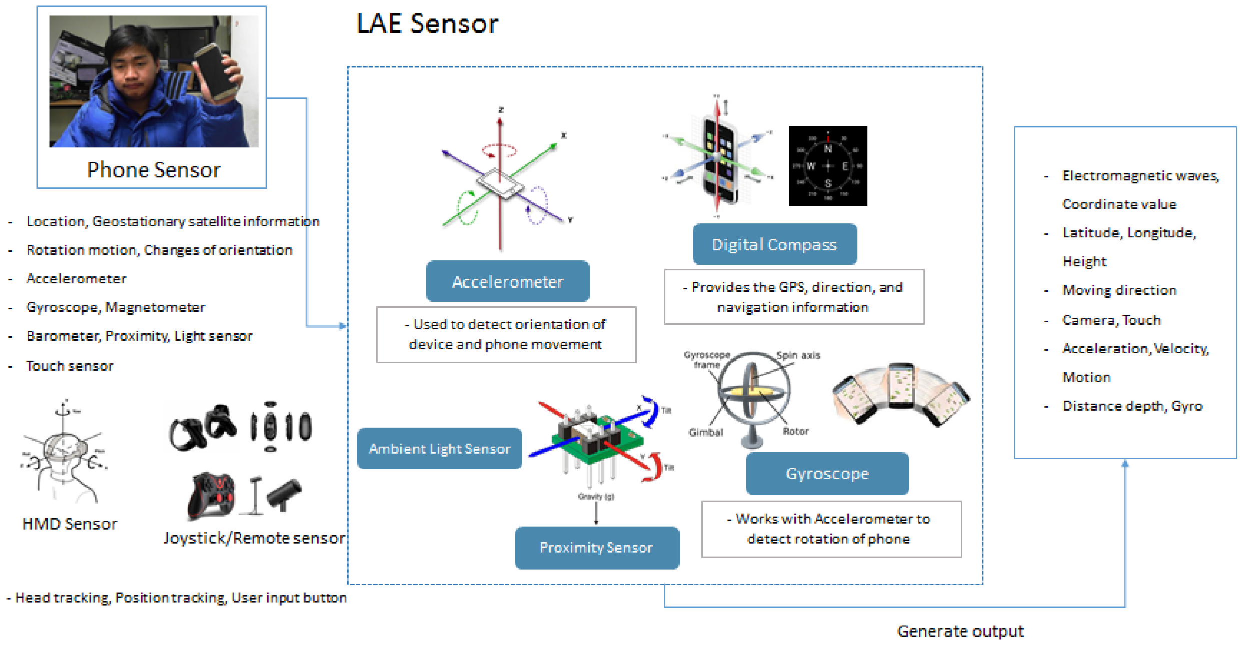


Figure 6. Examples of sensors for LAE activities

An HMD (head mounted display) sensor is also important for a LAE to be represented naturally in the virtual world. While a LAE is wearing an HMD device, he/she can see the real-like scenes of the virtual world and interact with virtual objects by using gesture front of depth or leap motion devices. There are three types of HMD devices, each providing different sensor information for a LAE:

First, a PC HMD is a desktop peripheral that acts as an external monitor. It provides the deepest and most immersive VR experience. It can track position and orientation smoothly, allowing for easy calculation of the user's position, then generate real time position movement in virtual space. Most PC HMDs come with input devices, such as a camera for tracking position, or a joystick for interacting with objects in the virtual world.

Second, a mobile HMD is a device that can be connected via smart phone in order to visualize the virtual reality, and can be customized using a built-in SDK. Mobile HMDs can provide orientation and head tracking sensing data. This type is supported strictly for smart phones.

Third, there are other, lower cost, devices that can view the stereo scenes of virtual reality, such as a Drop-in phone viewer. Drop-in phone viewers are supported by many smart phones using simple stereo rendering and accelerometer tracking. They can only track orientation and, therefore, do not provide an as smooth or immersive virtual reality experience as the first two.

Head tracking allows a LAE to feel present in a 360° scene due to sensors, such as gyroscope, accelerometer, and magnetometer.

5.3 Informational View

Sensors receive various types of signals in the physical world as inputs, and obtain sensing data related to the representation of a LAE in a MAR world.

The input and output of sensors related to a LAE in a MAR world are:

- Input: Physical world signals and/or device signals
- Output: Sensor data related to representation of the LAE

Physical sensor devices have a set of capabilities and parameters. Based on their properties, sensor devices generate wide-ranging types of sensing data, including camera intrinsic parameters (e.g., focal length, field of view, gain, frequency range, etc.); camera extrinsic parameters (e.g., position and orientation); resolution; sampling rate; skeleton; mono, stereo, or spatial audio; and 2D, 3D (colour and depth), or multi-view video. To provide the sensing data in a universally-consistent way, sensor output consists of <Identifier>, <Type>, and <Attributes>.

In order to composite a 3D virtual space and a LAE, consideration needs to be paid to the LAE sensing devices themselves. The standard can use the following types of sensing devices: general cameras such as Web cameras, and depth cameras such as depth-like devices. For general cameras, the sequence of RGB images will be captured and the image itself will be used as the sensing data. Depth cameras provide the following: an RGB color image, a 3D depth image, multi-array microphones, and a motorized tilt. The captured image and/or the depth image can be used to embed the LAE into the 3D virtual space.

Table 2. LAE capturer and LAE sensor input/output

- Input / Output of LAE Capturer

Types	Inputs	Output	Devices to be used
Video	Physical world including LAEs	Image/video (gray, color, depth) Camera information	RGB camera Depth camera
360 video		360^{o} image/video Camera information	360 camera
Stereo video		Stereo image/video Camera information	Stereo camera
Skeleton		Skeleton Camera information	Depth camera
Audio	Physical world	Audio signal (spatial or stereo) Audio Information	Microphone

- Input / Output for LAE Sensor

Device	Inputs	Output
Geographic coordinate system sensor	Location, geostationary satellite information	Electromagnetic waves, coordinate value, latitude, longitude, height
Gyro sensor	Rotation motion, changes of orientation	Electromagnetic waves, coordinate value, moving directions
Phone sensor	Accelerometer, gyroscope, magnetometer, barometer, proximity, light sensor, touch sensor	Electromagnetic waves, coordinate value, camera, touch, acceleration, access, motion
Wii remote/joystick	User input button function	Electromagnetic waves, coordinate value, distance depth, force
HMD (head mounted display)	Head tracking, position tracking, user input button	Electromagnetic waves, coordinate value, motion gyro, acess

6. Tracker and Spatial Mapper for a LAE

6.1 Overview

Figure 7 shows that a LAE, which is represented in the physical world, can be captured, and sensing data extracted, by a LAE capturer and LAE sensor. The sensing data that relates to the LAE from the LAE capturer and/or LAE sensor needs to be pre-processed in order to provide optimized sensing data for doing spatial mapping of the LAE in the MAR world. LAE capturer refers to physical world capturing devices, such as depth cameras, general cameras, and 360^{o} cameras. The video, image, skeleton, and audio information will be parsed to a pre-processing function to compose the filtering, colour conversion, background removal, Chroma-keying, and depth information extraction by using the technique of computer vision. The results of pre-processing will be transmitted to a LAE tracker. Furthermore, the LAE sensor for capturing the event of the LAE's motion, such as position, direction, geographic coordinate system sensing, time, and motion, will provide that sensing data to a filtering function. The filtering function is used for processing the sensing data and recognizing the target object. The raw signal for representing a LAE and target object specification data will be parsed to the LAE tracker. The LAE tracker can track a variety of information related to a LAE, such as real camera information and the LAE itself in the physical world [15].

Tracking data from the LAE Tracker, such as camera information (FOV, position, orientation, etc.) and LAE information (Chroma-keying results, skeleton, geographic coordinate system, etc.) in the physical world, will be parsed to a spatial mapper for mapping the LAE's spatial data. The role of the spatial mapper for a LAE is to provide spatial relationship information (position, orientation, scale, and unit) between the physical world and the world of the MAR scene by applying the necessary transformations for the calibration of the LAE. The tracking data is mapped with a spatial scene description to embed the LAE into the spatial scene, then the calibrated spatial information will be parsed to a scene compositing module.

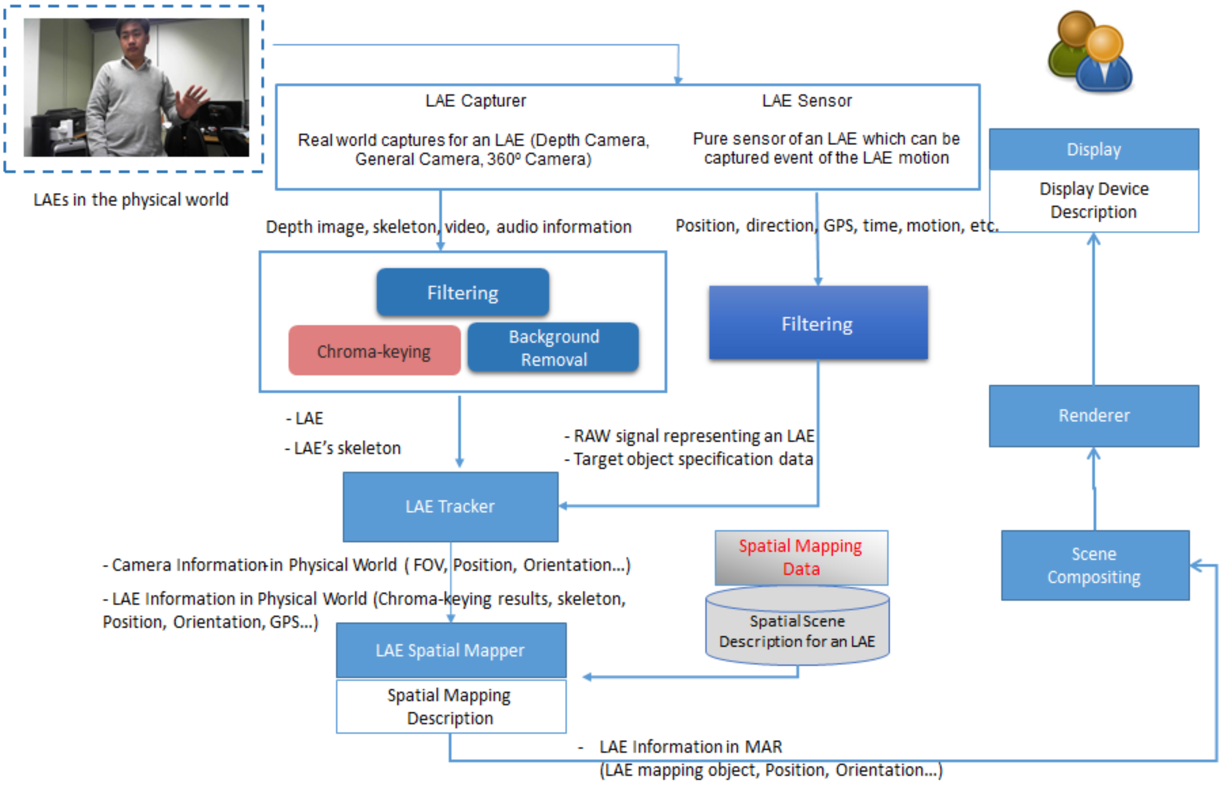


Figure 7. Tracker and spatial mapper of a LAE in MAR

6.2 Computational View

The work of the spatial mapper components can be done using information stored in the spatial scene for a LAE. Sensing data includes a coordinate system which refers to a LAE sensor that extracts sensing data from the physical coordinate system, the camera coordinate system, and the world coordinate system.

Figure 8 illustrates the mapping between the physical coordinate system and the world coordinate system, and details are provided in the following text:

Physical coordinate system refers to a coordinate system that enables locating a LAE, and which is controlled by a geographic coordinate system sensing device. This means that the coordinate system can be extracted from the physical scene through an implementation mechanism. The physical coordinate system retrieves ground coordinates, such as latitude/longitude or UTM coordinates, and can also measure ground distances and areas. The physical coordinate system can be applied to any kind of object or LAE that can be related to a geographical location.

Object LAE refers to the position, orientation, field of view (FOV), and spatial information of a LAE. The position is the location relative to the reference point of the object/LAE in the origin of the coordinate system. Orientation is the object's state of being oriented with a vector quantity and rotation of the LAE.

Physical camera coordinate system refers to a LAE capturer, which is the camera for capturing a LAE in the physical world. The physical camera also has its own coordinates, which can provide coordinate data to the spatial mapper.

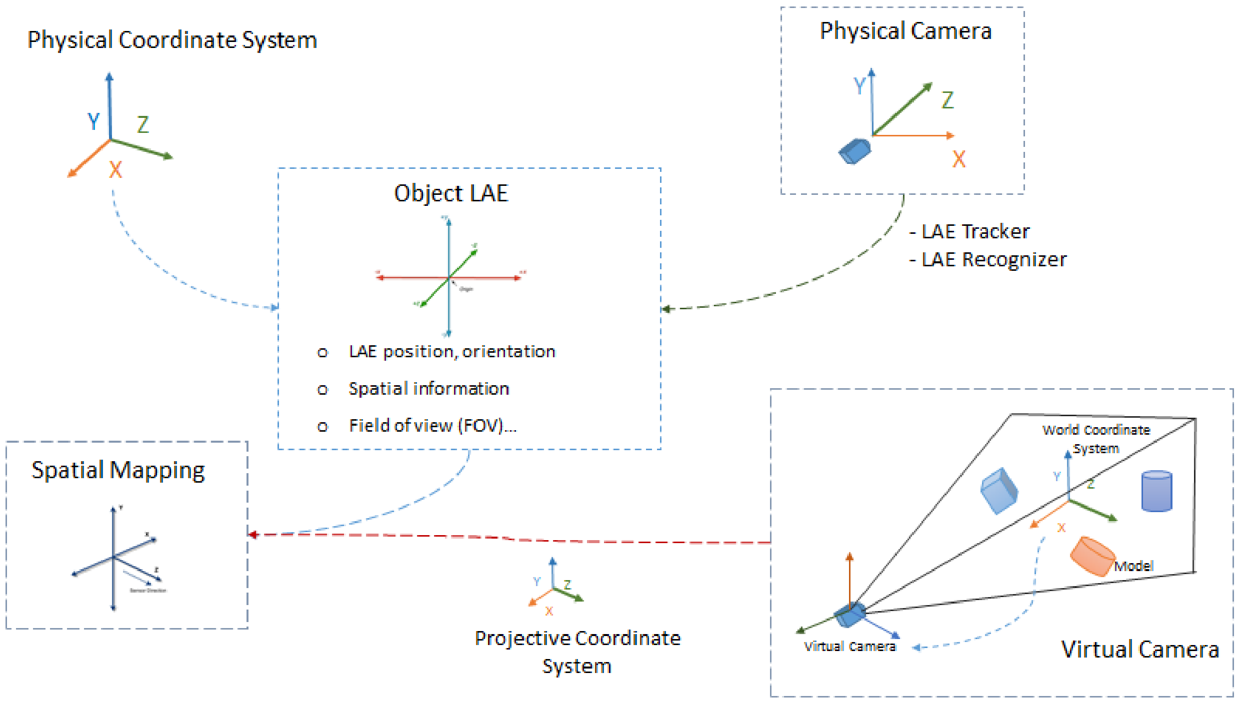


Figure 8. Mapping between physical and world coordinate systems

The coordinate system of a LAE starts from a model coordinate system that gets transformed into a world coordinate system, then into a virtual camera coordinate system, and finally into a projection coordinate system. The model coordinate system is the coordinate system where the entity's object model and LAE are initialized and created. It is a unique coordinate space of the model. Two distinct models, each with their own coordinate system, cannot interact with each other. Thus, there needs to be a universal coordinate system that allows any model to interact with any other. That universal system is called the world coordinate system. When interaction occurs, the coordinate system of each model and entities are transformed into a world coordinate system. The world coordinate system is then transformed into a coordinate system called the camera coordinate system.

The virtual camera coordinate system is what we see on the screen, and is related to a viewer which acts as a camera. A change in the camera's orientation and position changes what the viewer sees. The final transformation in OpenGL that converts three-dimensional scenery into a two-dimensional image is called the projective coordinate system. What is perceived on screen as three dimensional is just an illusion. It is simply a two-dimensional image making use of the projective coordinate system. These coordinate systems will be parsed to spatial mapping for further implementation.

6.3 Informational View

The LAE tracker is able to detect and measure changes in the properties of sensing data related to the representation of a LAE in a MAR world. Tracking information, such as position, orientation, location, geographic coordinate system, and so on, from the sensor will be tracked in the physical world.

- The input and output of the tracker are:
  - Input: sensing data related to the representation of the LAE
  - Output: Instantaneous values of the characteristics of the recognized target signals of the LAE

Table 3. Tracker categories

Dimension	Type	Types
Input	LAE capturer	• Camera information • 2D image/video of LAE • Object specification data • 3D primitives (points, lines, polygons, shapes) • 3D mode
	LAE sensor	• Sensing Information
Output	LAE capturer	• Video for LAE • Position, orientation, volume, location • Haptic (force, direction, etc.) • Aural (intensity, pitch, etc.)
	LAE sensor	Sensor information (coordinate value, latitude, etc.)

The spatial reference frames and spatial metrics used in a given sensor need to be mapped into that of the MAR scene so that the sensed LAE can be correctly placed, oriented, and sized. The spatial relationship between a particular sensor system and a MAR space is provided by the MAR experience creator and is maintained by the spatial mapper.

- The input and output of the spatial mapper are:
    - Input: Sensor identifier (SID) and sensed spatial information of the LAE
    - Output: Calibrated spatial information of the LAE for the given MAR scene

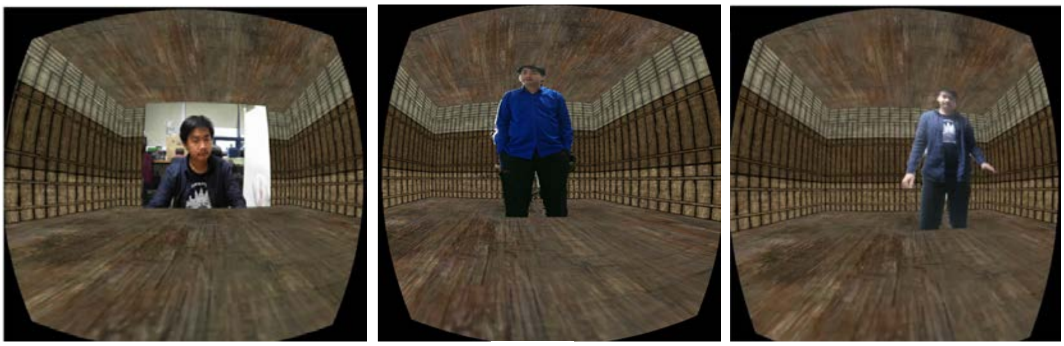
Table 4. Spatial mapper input/output

Dimension	Types
Input	• LAE identifier spatial data • LAE consecutive spatial data • Tracking spatial data • Spatial information
Output	• Calibrated spatial information • Audio (direction, amplitude, units, scale)

The notion of a spatial mapper can be extended to mapping other domains, such as audio (e.g., direction, amplitude, units, scale) and haptics (e.g., direction, magnitudes, units, and scale).

6.4 An Example of LAE Tracking and Spatial Mapping in MAR

Figure 9 (a) shows a LAE embedded into a MAR world without a Chroma-keying function to remove the background of the LAE in the physical scene. Figure 9 (b) shows that, after applying the Chroma-keying filter on the LAE video stream, the LAE can be embedded in the MAR world without its background. Figure 9 (c) shows a LAE moment in time in the MAR world. The LAE can be tracked from the physical world by the LAE capturer; and the position, orientation, FOV, coordinates, and other information of a LAE can be mapped into the spatial mapping. Spatial mapping is used to map the spatial data of a LAE and the scene description which contains the 3D virtual world.



(a) (b) (c)

Figure 9. An example of tracking and spatial mapping of a LAE in MAR

7. Recognizer and Event Mapper for a LAE

7.1 Overview

After obtaining sensing data from the LAE sensor, the data will go through to the LAE recognizer for recognizing events of LAE gestures, object collision, and so on.

The recognizer is a component that analyzes sensing data related to the representation of a LAE in a MAR world and produces MAR events and data activated from the LAE through comparison with a local or remote target signal (i.e., the target for augmentation) stored in the MAR scene.

7.2 Recognizer

Recognition can only be based on previously captured target signals. Both the recognizer and the tracker can be configured with a set of target signals provided by or stored in an external resource (e.g., a third party database server) in a manner consistent with the scene definition. The recognizer can be independently implemented and utilized. There are two types of data used by the recognizer: the output from the LAE capturer and the output from the LAE sensor.

Figure 10 shows an event of gesture of a LAE in the physical world captured by the LAE sensor and the LAE capturer. After filtering the LAE's raw data, the filtered data (position, orientation, direction, motion, etc.) will be parsed to the LAE recognizer for recognizing the specific target event. The LAE recognizer will be processed according to the event mapper and event description for the LAE. The event mapping data is stored in the event description for the LAE and can be used by the LAE recognizer according to the specific target and event_ID. The LAE's event information, such as event_ID, event_type, and event function, will be parsed to the event mapper for mapping the event. By analyzing the LAE information, the event mapper will produce the MAR event in the MAR scene. The MAR event results from the detection of a relevant condition from the LAE in the physical world and augmentation. The event result will be parsed to the scene compositing module and renderer in order to render the event function for display and UI.

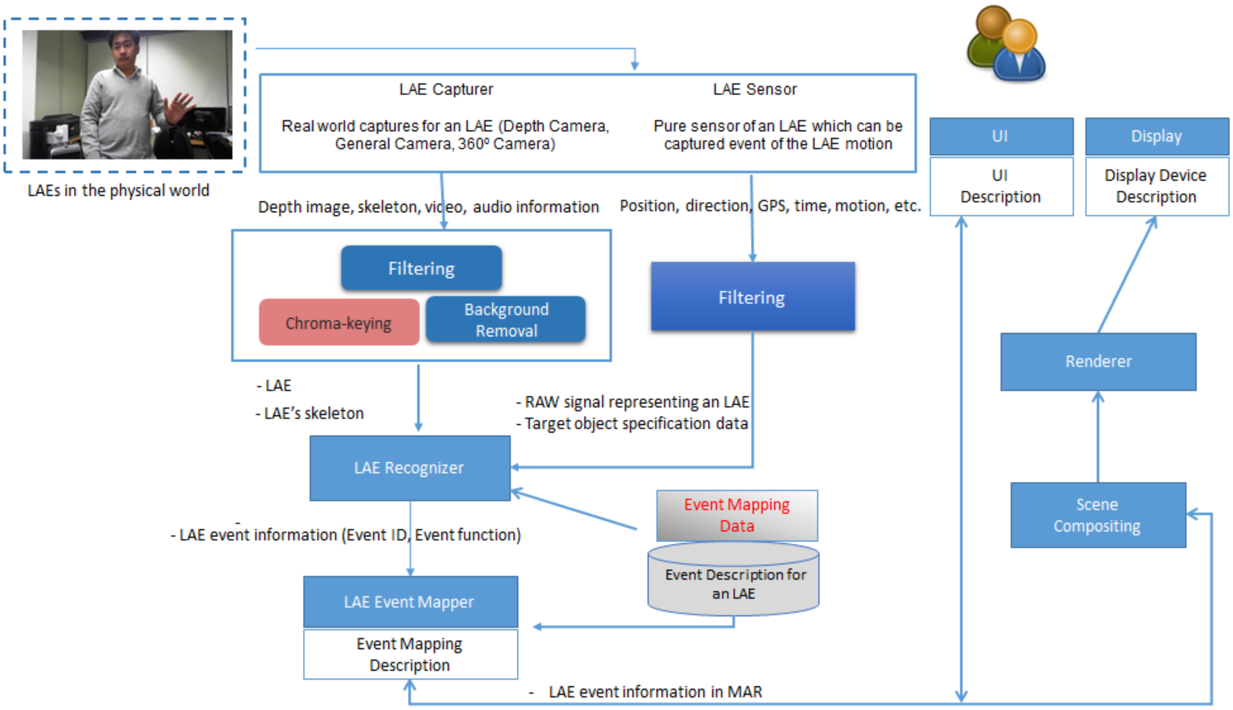


Figure 10. Recognizer and event mapper of a LAE in MAR

- The input and output of the recognizer are:
  - Input: Sensing data related to the representation of the LAE in the MAR world. The input data model of the recognizer is the output of the sensors. The other input to the recognizer, the target physical object data, should contain the following elements: First, it should have an identifier indicating the event when the presence of the target object is recognized. Second, the target physical object specification may include raw template files used for the recognition and matching process, such as image files, 3D model files, sound files, etc. Third, it may include a set of feature profiles. The types of features depend on the algorithms used by the recognizer. For instance, it could be a set of visual feature descriptors, 3D geometric features, etc.
  - Output: At least one event acknowledging the recognition, which identifies the recognized target and optionally provides additional information that should follow a standard protocol, language, and naming convention.

Table 5. Recognizer categories

Dimension	Type	Types
Input	Physical world	• Camera information • 2D Chroma-keying image • Audio • Gesture
	Sensor	• Sensor information
Output	Physical world	• Event indication only of the recognized object • Event by gesture and audio information
	Sensor	• Event indication only of the recognized object

7.3 Event Mapper

The Event mapper creates the relationship to the MAR event of a LAE that is obtained from the LAE recognizer and/or LAE tracker. It describes the MAR event that can occur due to the controlling event (gesture, movement, head tracking, sensor, etc.) and the condition specified in the MAR scene. In order to map the MAR event within the MAR scene, as well as the events identified and recognized by the LAE recognizer, event_ID, event_type, and eventDB (event database) are needed. The event refers to the sequence of the controlling event of a gesture by the LAE. The gesturing data will be recognized by the LAE recognizer to determine the type of event, and then the event function will be retrieved from the eventDB.

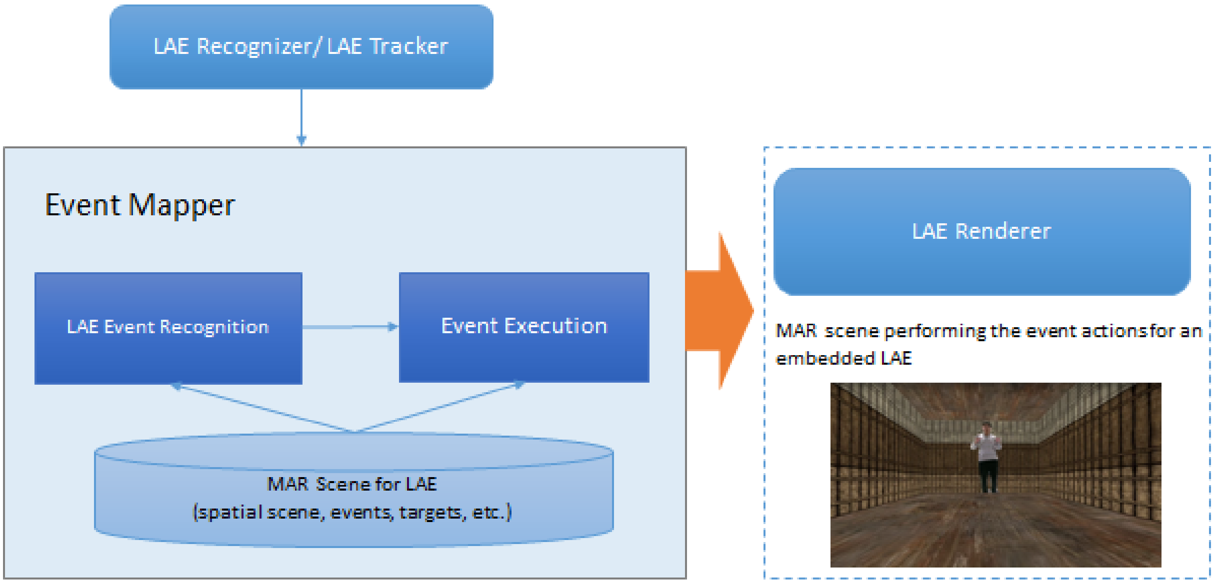


Figure 11. A procedure for controlling an event of a LAE in a MAR world

After retrieving the sensing data of the LAE, the event control model proposed in Figure 11 can be concretely developed with two modules, a recognition module and an execution module. Especially, the objective is to obtain faster and more accurate sensing data from the LAE itself and its handling devices in real time. If a hand gesture is used to control the events of a LAE, a depth camera, which allows skeleton images of one or two LAEs to be tracked, can be employed as a motion-sensing input device. Position information for the left and right hands can be obtained via depth. A position in 3D real space is represented by three Cartesian coordinates (x, y, depth) at each joint of the skeleton. An acquired position is first filtered to suppress image noise and then reprocessed with depth calibration and depth cueing to reduce image flickering. Depth supports some filtering features, but the resulting object boundaries remain very unstable and there is flickering. After all filtering processes have been carried out, hand gestures will be recognized by the recognition module, and the recognized event will be executed by the execution module in a virtual space, according to the gestures.

The event relationship between a particular recognition system and a target scene is provided by the MAR experience creator, and is maintained by the event mapper.

- The input and output of the event mapper are:
  - Input: Event identifier and event information
  - Output: Translated event identifier for the given MAR scene

Table 6. Event mapper input/output

Dimension	Types
Input	• Event Information • Event identifier (virtual camera control, virtual object control, AR content control)
Output	• Translated event identifier for the given MAR scene • Interaction of a virtual object event

7.4 Event Execution

A detected event is transferred in the event class *EventClass*, in an execution module. If the detected event is one of the events defined in the MAR system, the event corresponding to the recognized event will be applied to the object in the 3D virtual space by the event executor *EventExecutor*. In other words, the object in the 3D virtual space will be manipulated according to the corresponding event function.

Figure 12 shows the structure for executing an event to control an object in a 3D virtual space. The class *EventExecutor* employs registered object-control functions to control objects in the 3D virtual space by analyzing events in *EventClass*. If a virtual object can be controlled by an event, the class *EventExecutor* is set as a member of the object instance, a callback function is created to be executed on the gesture event for the object, and the generated function is linked to a corresponding gesture delegate function in *EventExecutor* based on the event type. Then, when an event occurs, the MAR system executes the function *executeCommand*, to which the event is passed as an *Event* instance, and executes the generated callback function via the corresponding delegate function in *EventExecutor*. As a result, a LAE can interact with 3D virtual objects in a MAR environment just as it can with real-world objects.

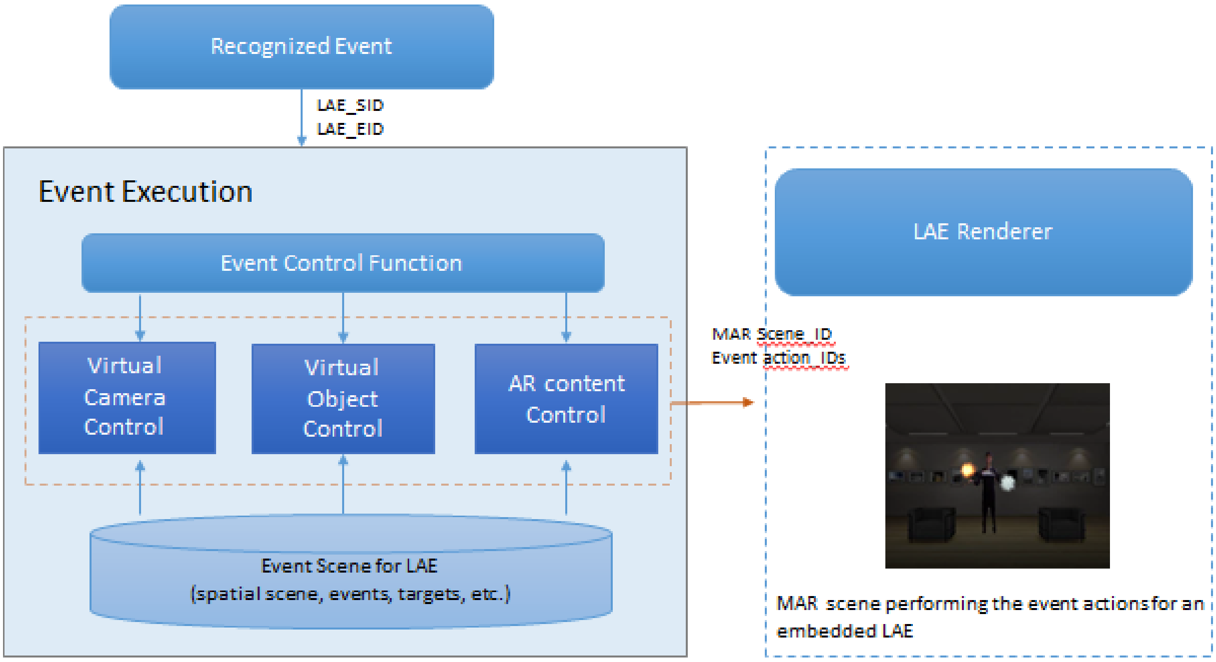


Figure 12. Execution structure of events performed by a LAE

7.5 Examples of LAE Recognizing and Event Mapping in MAR

Figure 13 shows the interaction of a LAE with virtual objects in a MAR scene [16-17]. Examples of interactions include dancing on a virtual Dance Dance Revolution (DDR) [18] stage, dodging a piece of wood, and walking on a balance beam. If the LAE follows prescribed conditions accurately, they win the game, otherwise they lose. In the first example, the man moves (dances) in front of the camera, the motion is captured, and the music plays according to the motion of his feet. In the second example, the piece of wood is a virtual object in the 3D MAR world, and the man tries to dodge the object. If he is unsuccessful, a collision between his head and the virtual object will occur. In the third example, the man is walking in the physical world but it appears as if he is crossing a balance beam in the MAR scene. If he falls, an event will occur.

Figure 13. Interaction of a virtual object in a MAR scene and event IDs of a LAE

Figure 14 shows events of a LAE in a MAR world based on recognition of gestures and event IDs [19]. The man is embedded into the MAR world by applying the spatial mapper and filtering with the Chroma-keying function to remove the background. The man can use his hands to create and control the events in the MAR world by doing hand gestures. The gestures are recognized by the LAE recognizer. The event IDs are retrieved from the eventDB and mapped with a module/function in the MAR scene by event mapping. An event occurs based on an event_ID, such as lifting the right hand creates fire, or lifting two hands creates both fire and ice, or holding the hands together makes the fire disappear.

Lift a right hand — Lift two hands — Hold two hands

Figure 14. Augmenting a virtual object based on event IDs of a LAE

8. Scene Representation for a LAE

8.1 Overview

MAR scene refers to a scene that represents a virtual scene and placeholders. It serves as the implementation structure that combines the physical world scene/objects and the virtual scene/objects. It is the observable spatiotemporal organization of physical and virtual objects that have been tracked and recognized by the LAE tracker and LAE recognizer functions. The scene representation can be based on several proposed AR related formats so far [20-25].

An event of a LAE can occur based on the LAE's own actions, as well as based on output obtained from sensing interface devices handled by the LAE. The LAE will be captured from general cameras and/or depth cameras. Its actions can be obtained by recognizing LAE sensing data. Meaningful events to be used in the MAR world can be classified into one of two types: sensing data by the LAE itself, or sensing data of devices handled by the LAE. The defined events will be stored in the eventDB.

- Sensing outputs captured from a "capturer" for the LAE
  - Gestures – hands, fingers, head, body
  - Facial expressions
  - Speech and voice
- Sensing outputs captured from a "sensor" for the LAE
  - AR marker
  - Global Positioning System (GPS[2])
  - Remote Wii motion data
  - Other sensing data by a smart device: three-axis accelerometer, magnetometer (digital compass)

Sensing data will be used to define the events of the LAE. We assume that a virtual camera, a virtual object, or an AR object in a 3D virtual space is controlled by human body gestures of the LAE. To define the gestures more efficiently, basic primitive postures of a human body are defined, as shown in Figure 15. The postures consist of position, vector, image, and skeleton. By detecting the postures of a human body and combining them, gestures can be defined. We define three common primitive gestures of a human body: linear, wave, clockwise/counterclockwise. A variety of events for a human body can be defined using these predefined primitive gestures.

A MAR scene describes all the information of a LAE that relates to a MAR world and consists of a virtual scene, sensing data, spatial scene, events, targets, etc. The MAR scene representation is the middleware of the spatial mapper and event mapper, which observes the spatial of physical objects, virtual objects, and events. A MAR scene has at least one physical object and one virtual object.

Figure 15 shows the scene representation for handling a LAE in a MAR world. After receiving sensing information from the sensor, the RAW signal representing the LAE and target object's specification data will be parsed to the LAE tracker and LAE

[2]The term GPS is specific to the United States' GNSS system, the NAVSTAR Global Positioning System. As of 2013, the (GPS) is fully operational GNSS.

recognizer. Thus, the LAE and physical world objects can be captured and detected. The LAE's information can be calculated and mapped in the virtual scene according to the LAE's position, orientation, etc. Calibration and spatial mapping between real and virtual objects are needed in a MAR scene. A physical world object is mapped to a placeholder in the MAR scene by, e.g., image feature, marker pattern definition, or geographic coordinate system, and provides the information data/method to recognize and track the object for semantic information as to what the object actually is. The video of the entire scene, including the background, will be processed in a background removal function to extract the LAE and its information. The LAE is embedded in the virtual scene via real time streaming, and the virtual objects of the virtual scene can be interacted with by the LAE's gesture/event according to the event of the LAE recognizer. When there are changes to a virtual object, MAR behavior will invoke the MAR event and the LAE recognizer will recognize then produce the event.

Sensor/device parameters are important for correctly presenting a MAR scene, e.g., the relationship of a physical world capture sensor parameter to the MAR scene viewpoint and image scale. The viewpoint and image scale must be changed according to the information captured by the physical world capture sensor.

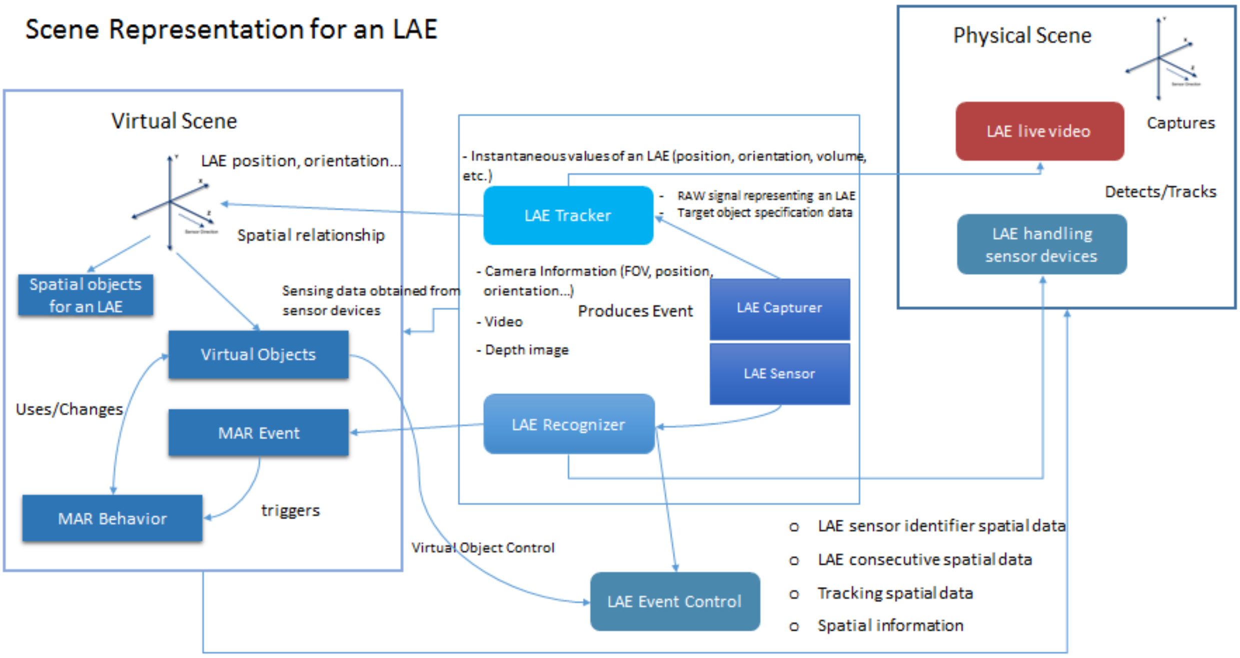


Figure 15. Scene representation for handling a LAE in MAR

8.2 Scene Description

MAR scene description will be used to perform more effective spatial mapping and event handling of a LAE in a MAR world. Two types of scene description will be provided.

First, spatial mapping is used for mapping spatial relationship information between a physical world and a MAR world. It describes how a LAE in the physical world can be mapped into a MAR system. LAE information provided by the LAE tracker and/or LAE recognizer will set a LAE ID and initialize the spatial mapping information. The spatial reference frames and spatial metrics used in a given sensor need to be mapped into the MAR scene so that the LAE mapping object can be correctly placed, oriented, and sized. The spatial mapping information can be modelled as a table, with each entry characterizing the translation process from one aspect (e.g. lateral unit, axis direction, scale, etc.) of the spatial property of the sensor to the given MAR scene [5].

Table 7. Scene description for spatial mapping of a LAE

LAE ID	Initial spatial mapping information	LAE mapping object	Spatial mapping function
LAE_ID	Position, orientation, etc.	Virtual 2D/3D object	Spatial mapping from physical world to MAR world

Second, event mapping describes how a LAE in a MAR world can produce an event, such as interaction, gesture, movement, voice recognition, etc. It creates an association between a MAR event obtained by the LAE recognizer and/or LAE tracker and a condition specified in the eventDB. For example, a LAE in a MAR world can use its hands to control/interact with virtual objects by doing hand gestures (palms up/down, hands clutched, hands raised, etc.). An event will occur when the LAE does the gesture or movement or handles a sensing device (e.g., smartphone, joystick, etc.). The sensor device handled by the LAE can generate the sensing data for creating an event in the event mapper. Event mapping matches the sensor_id and event_id which is retrieved from the eventDB. Then, it will know the type of event and call a function to be executed.

Table 8. Scene description for event mapping of a LAE

LAE ID	Event ID	Event Type	Event Function
LAE_id	Event_id	Event_type	Event call functions

9. Renderer

9.1 Overview

Rendering refers to the process of generating data from a 2D/3D model, updating a simulation, and rendering the presentation output for a given display device. After the processes of the spatial mapper and event mapper have completed, the data will be parsed to the renderer. Thus, the renderer has the job of rendering LAEs, virtual objects, and events to a display device.

Rendering of a LAE in a MAR system needs to be done based on the type of display, such as Web rendering to a browser, or PC, mobile, and stereo rendering. Stereo rendering has to do with the stereoscopic separation of the left and right eye. Rendering needs to be smooth and in real time, based on the user's HMD.

There are two issues when it comes to rendering. The first has to do with spatial mapping rendering, which refers to a LAE being embedded into a MAR scene. Based on the embedding process, the renderer needs to generate a high quality and smooth display. The second has to do with event mapping, which refers to the events of a LAE, such as natural movement rendering, gesture, voice recognition, and so on. The renderer needs to render the event perfectly in order to produce a quality, streamlined event. That is, when a LAE moves or does gestures, this should be reflected in the MAR world in a streamlined fashion. Furthermore, the renderer needs to control problems that might occur during LAE movement, such as moving to an unacceptable position and orientation, or gestures and movement being too slow, which may result in low quality rendering.

9.2 Computational View

A MAR system can specify various capabilities of the renderer, so a scene can be adapted and simulation performance can be optimized. The rendering of virtual reality, a stereoscopic HMD, and a mobile device might require different rendering performance. Multimodal output rendering might necessitate careful millisecond-level temporal synchronization. The output is a visual, aural, and/or haptic stream of data (such as a video frame, stereo sound signals, etc.) to be fed into a display device.

9.3 Information View

- The input and output of the renderer are:
  - Input: MAR scene graph data
  - Output: Synchronized rendering output (e.g., visual frame, stereo sound signals, motor commands, etc.)

The renderer can be categorized in the following way:

Table 9. Renderer categories

Dimension	Types			
1. Modality	Visual	Aural	Haptics	Others

10. Display and UI

Display refers to a hardware component that produces the actual presentation of a MAR system for representing a LAE to the end user. Displays include monitors, HMDs, projectors, scent diffusers, haptic devices, sound speakers, etc. The display needs to meet specific requirements in order to provide a good quality display to the end user. An actuator is a special type of display that does not directly stimulate the end user's senses but rather produces a physical effect in order to change properties of physical objects or the environment.

A UI is a component of hardware used to capture user interactions (e.g. touch, click) for the purpose of modifying the state of a MAR scene. A UI requires sensors to achieve this purpose. There are many types of UI sensors for capturing user interactions, such as gesture, voice, haptic, gyro, and so on. These sensors have a usage similar to that of LAE sensors. However, the difference is that, for UI sensors, the only physical object sensed is the user.

There are two kinds of events that can be generated:

- A LAE can generate events by itself – the LAE uses gestures, voice, head movements, etc.
- A LAE can handle a device for generating events – the LAE can use a smartphone, haptic, joystick, etc.

- The input and output of the display are:
  - Input: Rendered signals
  - Output: Display output
- The input and output of the UI are:
  - Input: User action, gesture, voice, interaction
  - Output: UI event

11. Extensions to Virtual LAE

Virtual LAE refers to a virtual representation of a LAE which is captured by 3D capturing technology and that can be reconstructed in a MAR world. The virtual LAE appears as a real person that can be communicated with, though it is not. Figure 16 shows a virtual LAE restricted to and communicating in a MAR world. The virtual person is reconstructed and displayed in the MAR world as a virtual object. Recently, a new type of 3D capture technology is appearing so that it allows a high-quality 3D model of a person to be reconstructed, compressed, and transmitted anywhere in the world in real time [4]. When combined with a mixed reality display, this technology allows a user to see, hear, and interact with a remote participant in 3D as if they are actually present in the same physical space. Communicating and interacting with a remote user becomes as natural as face-to-face communication.

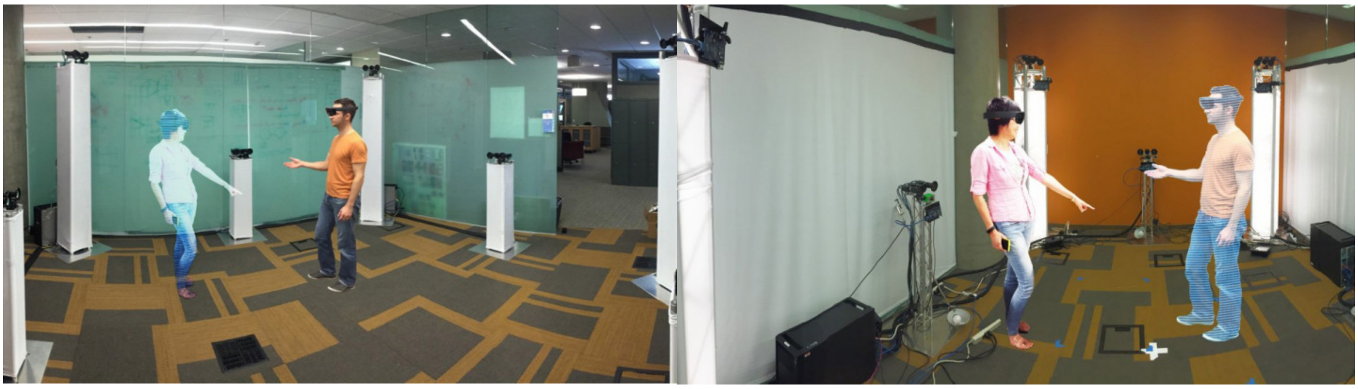


Figure 16. A virtual LAE restricted to and communicating in a MAR world [9]

An HMD device is required to view the hologram in real time, and a room surrounded with 3D cameras is necessary to create the hologram. The virtual LAE is not viewable without wearing an HMD device, and it is not possible to make eye contact with someone who is wearing the device.

A second type of virtual LAE refers to a virtual character (avatar) that is represented in a MAR world, as shown in Figure 17. This kind of virtual LAE can be used instead of a LAE in a physical scene. However, the interaction and motion of the virtual character must be accomplished by LAE gestures in the physical scene. The avatar can be represented along with other avatars, and the avatars can be made to communicate with each other. An avatar can be constructed with a 3D model, and points, like a LAE skeleton, need to be defined in order to produce animation and gestures. The virtual character and movements must be transmitted in real time using a LAE and movements in the physical scene, such as gestures of the hand, foot, head, and mouth.

Figure 17. Virtual character representation in a MAR world [19]

12. System Performance

System performance for a MAR LAE is an important part of the user's experience. The performance of a MAR LAE system can be evaluated in measurable and technical terms. Performance is based on sensor quality, quality of LAE expression, processing speed, response time, latency, dirty chroma-keying results, frame rate, and so on. Sensor refers to a LAE sensor, which is a sensor for capturing a LAE. The accurateness of the sensor affects the accuracy and quality of the LAE's information. The expression of the LAE can also be affected by system performance related to applying a Chroma-keying function, background removal, natural movement, and degree of freedom.

Latency can be measured as the total time needed to process the target object and produce the augmentation, i.e. the time delay between the cause and the effect of a physical change in the system being observed. It is a result of the limited velocity with which any physical interaction can take place. For example, the physical world LAE moves to the left, but the system update is slow and the LAE still shows at the right.

Augmentation precision can be measured as the error between the virtual camera parameters estimated by the tracker and the correct ones, or as the distance (in pixels) and angular distance (in degrees) between where the virtual object is displayed and where it should be displayed.

Operating conditions affecting performance may include lighting conditions, the mobility of the target object, sensing distance, orientation, etc.

Response time, which is the total amount of time it takes to respond to a request for any functionality, such as correlation detection, object control, 3D display rendering, etc., is another consideration. There are three types of response time: The first, service time, refers to how long the MAR system takes to respond to a request. The second, waiting time, refers to how long a request has to wait for another request queued ahead of it before it can run. The third, transmission time, refers to how long it takes to process the request and respond back to the requestor. Examples of requests include gestures for interacting with objects, a LAE's movement (updating position and movement in virtual space), etc.

13. Safety

A LAE in a MAR world can be presented with various safety issues, as it is a human being acting in the physical world. Thus, a LAE's attention needs to be focused. LAE safety guidelines are necessary to ensure that LAE in a MAR world content includes components for safeguarding the LAE during system operation. These safety guidelines can be used to reduce the risk for a LAE by considering the following:

- Dangerous obstructions that could lead to injury during the LAE's performance in the MAR world
- Any information that needs to be encrypted for security reasons
- Personal privacy of the LAE and potential exposure of personal information to unauthorized systems or third parties via a sensor/camera being out of scope, including authentication identity, system access to a LAE's personal data, etc.
- Due consideration of the physical world situation during LAE movement in the MAR world
- Virtual reality sickness from wearing an HMD
- Wearing an HMD device and being blind to potentially dangerous objects in the vicinity
- Avoiding quick acceleration or deceleration of camera movements, and using a constant velocity instead
- Keeping the frame rate up (less than 30fps is uncomfortable)
- Intermittent disconnection of the network service, leading to false confidence in the currently presented information
- Avoiding the use of Depth of Field or Motion Blur post processing because of not knowing where the eyes will focus
- Avoiding sharp and/or unexpected camera rotations
- Avoiding brightness changes (use low frequency textures or fog effects to create smooth lighting transitions)

14. Conformance

Conformance for LAE representation in a MAR world is expressed around the aspects of how a LAE can be embedded into a MAR world and the implementation process related to the LAE. The conformance of LAE representation to this standard shall satisfy at least the following requirements:

- The key architectural components that shall be present for LAE representation are the following:
    - LAE capturer
    - LAE sensor
    - LAE tracker
    - LAE Spatial mapper
    - LAE recognizer
    - LAE Event mapper
    - Scene representation
    - Renderer
    - Display and UI
- The three types that shall be controlled for LAE representation are the following :
    - Camera control
    - Object control
    - Augmented object control
- The implementation of LAE representation in a MAR world shall conform to the concepts and architectural components for a LAE-MAR system as shown in Figure 3 in Section 4.
- The processing implementation of a LAE-MAR system shall contain the Chroma-keying, filtering, tracking, and recognizing functions of capturing and sensing LAE as specified in Sections 5, 6, and 7.
- The movement of a LAE in a LAE-MAR system shall be mapped, and moved naturally within a MAR world according to LAE spatial mapper as specified in Section 6.
- The interaction between a LAE and a MAR world shall be performed naturally within a MAR world according to LAE event mapper as specified in Section 7.
- The interfaces between the architectural components of a LAE-MAR implementation shall contain and carry the information specified in this standard. However, the specific content, format, data types, handshake, flow, and other implementation details are at the discretion of the given LAE-MAR implementation to meet its specific needs.
- The API for a LAE-MAR implementation shall conform to the concepts specified in this standard in order to ensure compatibility and software interface interoperability between LAE-MAR implementations can be accomplished at least at the abstract API level.

Annex A. Use Case Examples

A.1 3D Virtual Studio

There are many examples where a LAE can be embedded into a MAR world. The first example is of a virtual broadcast studio. Figures 18(a) and (b) show a LAE as an announcer integrated with two different virtual broadcast studios. In order to visualize the LAE in the studio, the MAR system captures the image, including the LAE, by a general camera, removes the background of the image by processing Chroma-keying for the captured image, and then renders the 3D virtual world in which the image (with the background removed) is inserted by performing texture mapping on a polygon. Here, the polygon is a movable region of the LAE.

(a) (b)

Figure 18. Two different virtual broadcast studios including a LAE

A second example, in Figure 19, is of a LAE walking into a 3D virtual gallery and taking a seat. Rendering of the LAE is done as described above.

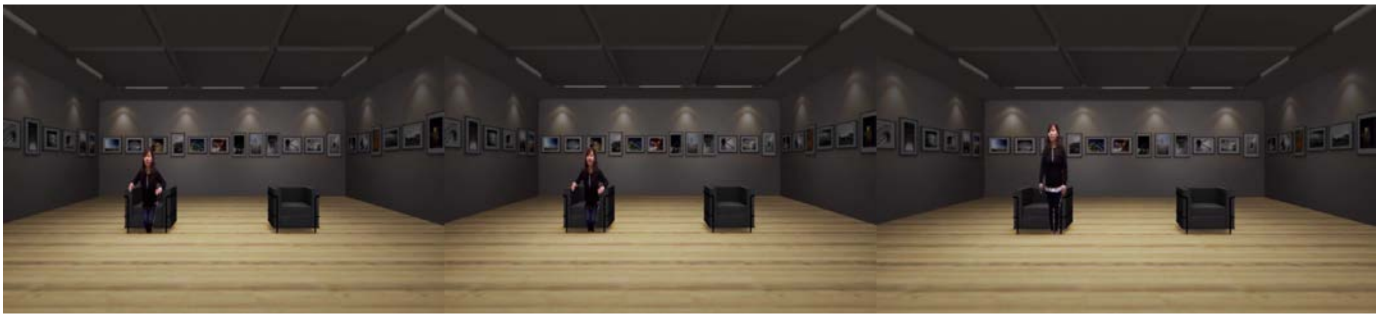


Figure 19. A LAE taking a seat in a 3D virtual gallery

A.2 Event Mapping of a LAE in a MAR world

There are many types of sensing interfaces between a LAE and a 3D virtual world. In this document, we consider hand gestures of the LAE itself. As proposed in the standard, three different types of content - virtual cameras, virtual objects, and virtual AR objects - in the 3D virtual world can be controlled through these predefined hand gestures.

Control of an interactive virtual camera by a LAE

A virtual camera in 3D virtual space can be controlled by the hand gestures of a LAE in a physical world. The movement and positions of the hands of a LAE can be obtained from a depth camera, which is a motion sensing input device used to capture position information for tracking a user's skeleton images. Positions are presented as the location (x, y, depth) at each joint point of the skeleton from the depth camera. Even if the depth input device processes filtering for the captured positions, flicker in the sequence of positions can still occur. Two operations, *depth calibration* and *depth cueing*, for the sequence of positions are done in a pre-processing step in a MAR system to reduce flicker.

To more accurately recognize the hand gestures of a LAE, the active space in which the hand gestures are performed is continuously monitored. A virtual camera will be moved according to the results of recognition of gestures in a valid subspace. Two hands, left and right, will be used to control the virtual camera in the MAR system. Twenty-four types of hand gestures can be created, based on four movement vectors - upper, lower, left, and right - of the two hands, combined. When the two hands are moved independently into place, 16 (= 4x4) gestures can be created; when one hand is fixed and the other is moved, 8 (= 2x4) gestures can be created; for a total of 24. In a MAR system, only six of the 24 gestures will be defined, as shown in Table 10.

TABLE 10. Types of hand gestures of a LAE for controlling a virtual camera in a MAR application

Gesture Type	Description
G1 (left rotation of a camera)	Action of moving the right arm to the right, while holding the left arm still
G2 (right rotation of a camera)	Action of moving the left arm to the left, while holding the right arm still
G3 (up rotation of a camera	Action of moving both the left and right arms upwards
G4(down rotation of a camera)	Action of moving both the left and right arms downwards
G5 (zoom in of a camera)	Actions of extending out both left and right arm.
G6 (zoom out of a camera)	Actions of bringing in both left and right arms

In order to simulate these six events using the hand gestures of a LAE, gesture events can be defined based on the movements of the two hands, as shown in Figure 20. For example, left rotation of a virtual camera will be performed by recognizing that the left hand moves to the left, while the right hand is fixed. For more accurate recognition, we assume that the center position of the virtual camera, which is always looking for the origin of the 3D virtual space with a fixed upper vector, can be placed at any point in the 3D virtual space. Under these circumstances, the center position will be transformed according to the gestures. To simulate a more natural movement of the camera, we have to consider the movements of the gestures. Assume that the current position of a hand is at any point in the 3D virtual space. When the hand is moved out of the designated circle, with its center and radius, the movement into one of four locations - left, right, upper, and lower, which are defined according to the relative position from the starting point, will be detected.

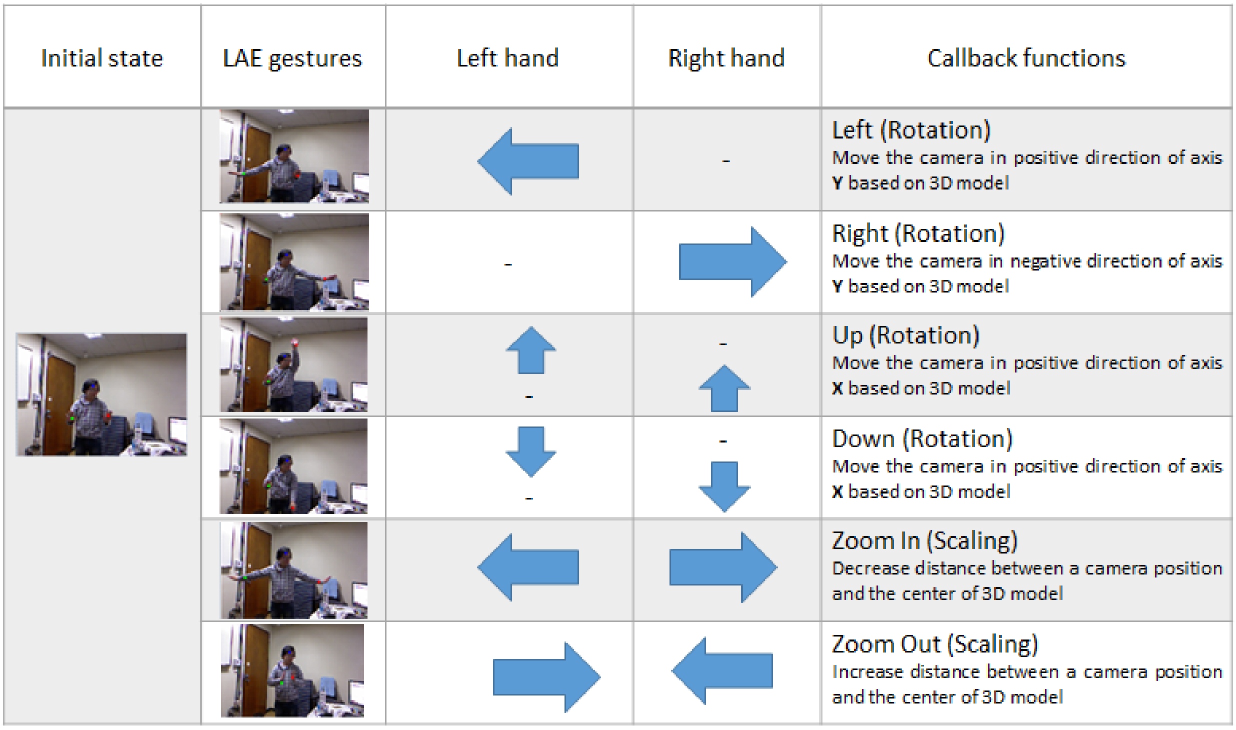
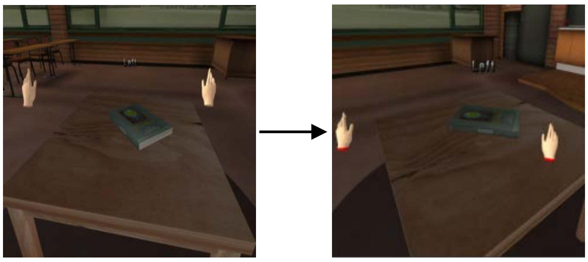
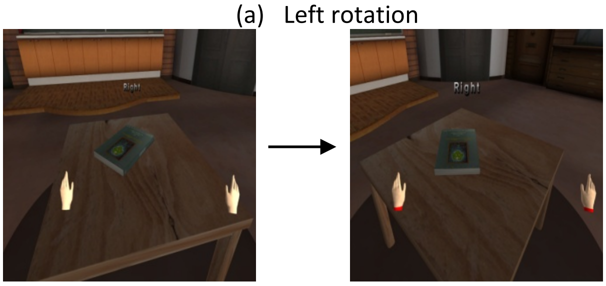


Figure 20. Gesture events for controlling a virtual camera in a MAR system [19]

A LAE performs one of six types of gestures to control a virtual camera in a 3D virtual space while viewing a rendering scene. Figure 21 (a) shows the result of applying a gesture to left rotate the center position of the virtual camera, after designating the initial position of the LAE. Similarly, Figure 21 (b), (c), (d), (e), and (f) show the results for right rotation, up rotation, down rotation, zooming in, and zooming out, respectively.

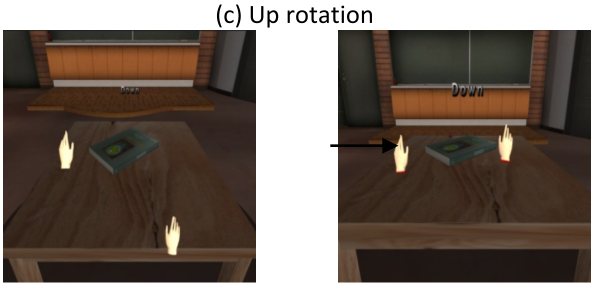


(a) Left rotation

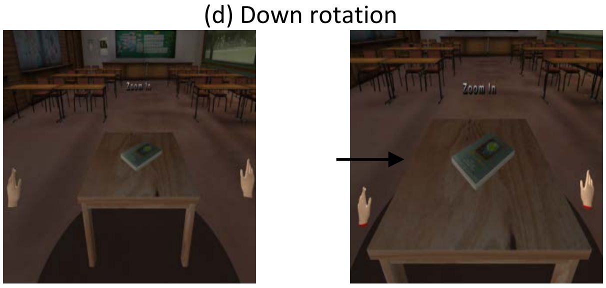


(b) Right rotation

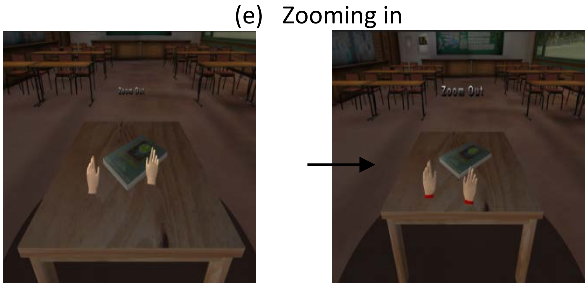
(c) Up rotation



(d) Down rotation



(e) Zooming in



(f) Zooming out

Figure 21. Examples of applying hand gestures to control a virtual camera in a MAR system [19]

A.3 Interactive Control of a Virtual Object in a MAR World by the Actions of a LAE

Hand gestures of a LAE are able to control virtual objects in 3D virtual environments in a manner similar to the virtual camera control described above. The hand gestures can be obtained by using a depth device. With the movement of two hands, seven gesture events can be defined, as shown in Figure 22. For example, right rotation of a virtual object will be performed by recognizing that the right hand moves to the right, while the left hand is fixed.

Initial state	LAE Gestures	Left hand	Right hand	Event_ID	Callback functions
		HG_NONE && HG_NONE		EC_NONE	None
		HG_PICKING | | HG_PICKING		EC_PICKING	Picking Pick an object in a MAR world
		HG_LEFT && HG_NONE && -		EC_LEFT	Left (Rotation) Move the camera in positive direction of axis Y based on 3D model
		HG_NONE | | HG_RIGHT - &&		EC_RIGHT	Right (Rotation) Move the camera in negative direction of axis Y based on 3D model
		HG_UP | | HG_UP - -		EC_UP	Up (Rotation) Move the camera in positive direction of axis X based on 3D model
		HG_DOWN | | HG_DOWN - -		EC_DOWN	Down (Rotation) Move the camera in positive direction of axis X based on 3D model
		HG_LEFT && HG_RIGHT		EC_ZOOMIN	Zoom In (Scaling) Decrease distance between a camera position and a center of 3D model
		HG_RIGHT && HG_LEFT		EC_ZOOMOUT	Zoom Out (Scaling) Increase distance between a camera position and a center of 3D model

Figure 22. Gesture events for controlling a virtual object in a MAR system [19]

Figure 23 shows a simulated replacement of a flat tire on a virtual car with a spare tire from the car's trunk. Figure 23 (a) shows the initial 3D virtual space in which the car is located, together with flesh-colored virtual left and right hands ready to perform gestures to control the car (virtual object).

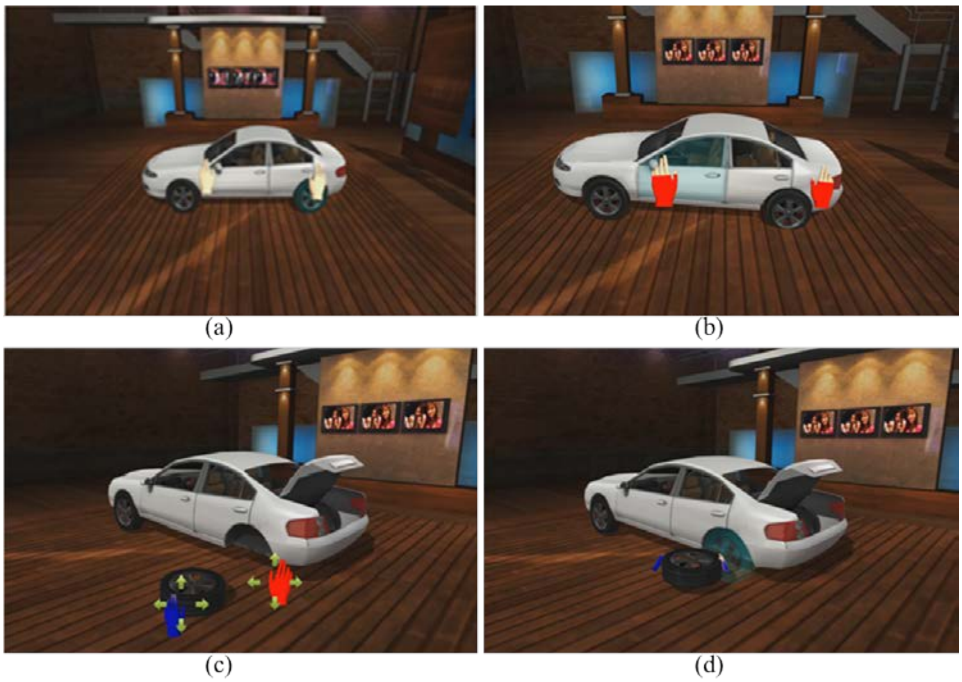


Figure 23. Gesture controlling and interacting with virtual objects in a MAR world

When the LAE's hand is located in the inactivation area, or when no gesture is being made, the virtual hand remains flesh colored. If the LAE's hand enters the activation area, the virtual hand begins to turn red. When the virtual hand turns completely red and is surrounded by four arrow keys, this indicates that the gesture recognition system has begun to recognize the hand gesture, as shown in Figure 23 (c). If the LAE's hand enters an object-selection area, the virtual hand begins to turn blue. When the virtual hand turns completely blue and is surrounded by four arrow keys, this indicates that the virtual object in the vicinity of the virtual hand is selected and is ready to be manipulated via real-world hand gestures. To replace the flat tire, we need to remove the flat tire, open the trunk, extract the spare tire, and set it on the ground. In Figure 23 (d), the blue hands hold the spare tire and push it closer to the car. The tire is then set upright with the left hand and attached to the wheel of the car. Thus, a virtual flat tire on a virtual car in a virtual space is replaced with a virtual spare tire from a virtual trunk. Figure 24 (e) shows an example of opening a door on the car and taking a seat, and then, in Figure 24 (f), controlling the audio system while seated. Figure 24 (g), (h), and (i) show an example of simulated engine disassembly.

(e) **(f)**

(g) (h) (i)

Figure 24. Examples of controlling virtual objects by hand gestures of a LAE in a MAR system

A.4 Augmenting a Real Object with Special Effects

Gestures of a LAE are able to augment virtual objects in 3D virtual environments in a manner similar to the other operations described above. The gestures can be obtained by using a depth device. Three gesture events can be defined with the movement of two hands and two arms, as shown in Figure 25. Figure 26 shows examples of using these three gestures. To augment the scene with a water animation, the left arm is bent and the left fingers are spread. Similarly, to augment with a fire animation, the right arm is bent and the right fingers are spread. The animations will continue until the disappearing gesture is recognized - in this example, by bringing the hands together.

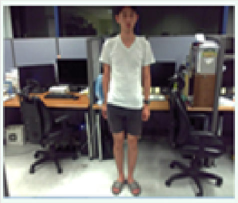
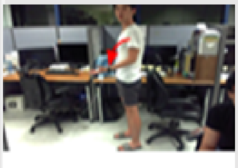
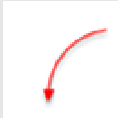
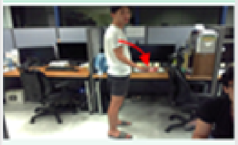
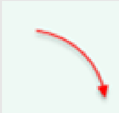
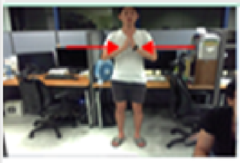
Initial status	Gesture operation	Left hand and arm	Right hand and arm	Function
			-	Generates a water animation augmentation by bending an left hand
		-		Generates a fire animation augmentation by bending an right hand
				Augmentation will disappear by bringing the two hands together

Figure 25. Gesture events for controlling an augmented object in a MAR system

(a) (b) (c)

Figure 26. Examples of augmenting real objects with special effects

A.5 3D Virtual Conference

Virtual conferencing and virtual presentations are other applications of a MAR system. LAEs can hold a conference or deliver a presentation in a 3D world. They can broadcast to join the conference in real-time, and embed themselves (as LAEs) into the 3D virtual wold. Multiple LAEs are able to move, communicate, open presentation files, watch videos, and interact with virtual 3D objects in the 3D virtual conference room.

Figure 27. Virtual conference and presentation in a MAR world

Bibliography

[1] Milgram, P., Takemura, H., Utsumi, A., and Kishino, F., Augmented reality: A class of displays on the reality-virtuality continuum, Proc. of Telemanipulator and Telepresence Technologies, 1994, pp. 2351–34.

[2] Azuma, R., A survey of augmented Reality, Presence: Teleoperators and Virtual Environments, 6 (4), 1997, pp. 355 - 385.

[3] Chroma key – https://en.wikipedia.org/wiki/Chroma_key.

[4] Jolesz, F., Image-guided procedures and the operating room of the future, Radiology, 204(3), 1997, pp. 601-12.

[5] Billinghurst, M., Kato, H., and Poupyrev, I., The MagicBook: A transitional AR interface, Computers and Graphics, 25 (5), 2001, pp. 745-753.

[6] Fuchs, H., Livingston, M., Raskar, R., Colucci, D., Keller, K., State, A., Crawford, J., Rademacher, P., Drake, S., and Meyer, A. Augmented reality visualization for laparoscopic surgery, Proc. of Intl. Conference on Medical Image Computing and Computer Assisted Intervention, 1998, pp. 934-943.

[7] Mixed Reality VR testing – Bowslinger & Holoball https://www.youtube.com/watch?v=Kw5yPyv9O9E.

[8] Kato, H. and Billinghurst, M., Marker tracking and HMD calibration for a video-based augmented reality conferencing system, Proc. of Intl. Workshop on Augmented Reality, 1999, pp. 85-94.

[9] Microsoft Holoportation – https://www.microsoft.com/en-us/research/project/holoportation-3/.

[10] OpenCV, http://code.opencv.org, 2017

[11] WebVR, http://www.webvr.info, 2017.

[12] WebVR Concept, https://developer.mozilla.org/en-US/docs/Web/API/WebVR_API/WebVR_concepts, 2017.

[13] Chheang Vuthea, Ga-Ae Ryu, Sangkwon Jeong, Gookhwan Lee, Kwan-Hee Yoo, A Web-based System for Embedding a Live Actor and Entity using X3DOM, Korean society of broadcast engineers, 2016, pp. 1-3.

[14] Jeon, S. and Choi, S., Real stiffness augmentation for haptic augmented reality, Presence: Teleoperators and Virtual Environments, 20(4), 2011, pp. 337-370.

[15] Klein G. and Murray, D., Parallel tracking and mapping for small AR workspaces, Proc. of Intl. Symposium on Mixed and Augmented Reality, 2007, pp. 1-10.

[16] Thomas, C., Close, B., Donoghue, J., Squires, J., De Bondi, P., Morris, M., and Piekarski, W., ARQuake: An outdoor/indoor augmented reality first person application, Proc. of Intl. Symposium on Wearable Computing, 2000, pp. 139-146.

[17] Lavric, T., Scurtu, V., and Preda, M. Create and play augmented experiences, Presentation from the 104th MPEG meeting, Inchon, 2014.

[18] Johanna H., International Survey on the dance dance revolution game, Retrieved January 29, 2015

[19] Jong-Oh Kim, Mihye Kim, and Kwan-Hee Yoo, Real-Time Hand Gesture-Based Interaction with Objects in 3D Virtual Environments, International Journal of Multimedia and Ubiquitous Engineering, 2013, pp. 339-348.

[20] MPEG ARAF: Augmented magazine and printed content, www.youtube.com/watch?v=CNcgxEOt_rM, 2015.

[21] ISO/IEC 23000-13, MPEG Augmented reality application format, 2014.

[22] Open Geospatial Consortium (OGC), Keyhole Markup Language (KML), www.opengeospatial.org/standards/kml, 2015.

[23] Mobilizy, ARML (Augmented Reality Markup Language) 1.0 Specification for Wikitude, www.openarml.org/wikitude4.html, 2015.

[24] Hill, A., MacIntyre, B., Gandy, M., Davidson B., Rouzati, and H. Khamra: An open KML/HTML architecture for mobile augmented reality applications, Proc. of Intl. Symposium on Mixed and Augmented Reality, 2010, pp. 233-234.

[25] ISO/IEC NP 19710, JPEG AR, 2014.