International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

H.704

(08/2020)

SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS

IPTV multimedia services and applications for IPTV – General aspects

Enhanced user interface framework for IPTV terminal device – Gesture control interface

Recommendation ITU-T H.704

ITU-T H-SERIES RECOMMENDATIONS

AUDIOVISUAL AND MULTIMEDIA SYSTEMS

CHARACTERISTICS OF VISUAL TELEPHONE SYSTEMS	H.100–H.199
INFRASTRUCTURE OF AUDIOVISUAL SERVICES	
General	H.200–H.219
Transmission multiplexing and synchronization	H.220–H.229
Systems aspects	H.230–H.239
Communication procedures	H.240–H.259
Coding of moving video	H.260–H.279
Related systems aspects	H.280–H.299
Systems and terminal equipment for audiovisual services	H.300–H.349
Directory services architecture for audiovisual and multimedia services	H.350–H.359
Quality of service architecture for audiovisual and multimedia services	H.360–H.369
Telepresence, immersive environments, virtual and extended reality	H.420–H.439
Supplementary services for multimedia	H.450–H.499
MOBILITY AND COLLABORATION PROCEDURES	
Overview of Mobility and Collaboration, definitions, protocols and procedures	H.500–H.509
Mobility for H-Series multimedia systems and services	H.510–H.519
Mobile multimedia collaboration applications and services	H.520–H.529
Security for mobile multimedia systems and services	H.530–H.539
Security for mobile multimedia collaboration applications and services	H.540–H.549
VEHICULAR GATEWAYS AND INTELLIGENT TRANSPORTATION SYSTEMS (ITS)	
Architecture for vehicular gateways	H.550–H.559
Vehicular gateway interfaces	H.560–H.569
BROADBAND, TRIPLE-PLAY AND ADVANCED MULTIMEDIA SERVICES	
Broadband multimedia services over VDSL	H.610–H.619
Advanced multimedia services and applications	H.620–H.629
Ubiquitous sensor network applications and Internet of Things	H.640–H.649
IPTV MULTIMEDIA SERVICES AND APPLICATIONS FOR IPTV	
General aspects	**H.700–H.719**
IPTV terminal devices	H.720–H.729
IPTV middleware	H.730–H.739
IPTV application event handling	H.740–H.749
IPTV metadata	H.750–H.759
IPTV multimedia application frameworks	H.760–H.769
IPTV service discovery up to consumption	H.770–H.779
Digital Signage	H.780–H.789
E-HEALTH MULTIMEDIA SYSTEMS, SERVICES AND APPLICATIONS	
Personal health systems	H.810–H.819
Interoperability compliance testing of personal health systems (HRN, PAN, LAN, TAN and WAN)	H.820–H.859
Multimedia e-health data exchange services	H.860–H.869
Safe listening	H.870–H.879

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T H.704

Enhanced user interface framework for IPTV terminal device – Gesture control interface

Summary

Recommendation ITU-T H.704 defines the general requirements, functional elements and interfaces supporting enhanced capability of user interaction by gesture recognition and controlling over Internet protocol TV (IPTV) terminal devices, based on the enhanced user interface (UI) framework defined in Recommendation ITU-T H.703. Those functional elements are described in the gesture controlling enabler and gesture recognition enabler defined in this Recommendation. Moreover, the procedures of interaction between the gesture recognition device and the gesture-controlled device are defined with the recommended information used in the interaction.

This Recommendation enables the gesture controlling feature in the enhanced IPTV user interface defined in Recommendation ITU-T H.703. With those features, users can control the operation of IPTV applications in an IPTV terminal device in a convenient, natural and comfortable way.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T H.704	2020-08-13	16	11.1002/1000/14343

Keywords

Enhanced user interface, gesture control, gesture control enabler, gesture interface, gesture recognition, IPTV.

---

* To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Recommendation ITU-T H.704

Enhanced user interface framework for IPTV terminal device – Gesture control interface

1 Scope

This Recommendation describes the gesture event features, general requirements and the functionalities of the framework to support enhanced user interface for an Internet protocol TV (IPTV) terminal device. This Recommendation focuses on gesture control in a high-level approach.

The purpose of the enhanced user interface (UI) framework is to define the functional elements supporting enhanced capability of user interaction over an IPTV terminal device regarding a gesture control interface.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T H.703] Recommendation ITU-T H.703 (2016), *Enhanced user interface framework for IPTV terminal devices.*

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following term defined elsewhere:

3.1.1 video sensor [b-ITU-T H.742.0]: A device that has image processing ability, such as image compression, image manipulation with the video data captured by a camera. A video sensor can generate video sensor information alone, by processing video data when it has enough computational power. Otherwise, some functions of the video sensor are processed through external systems and the video sensor generates intermediate video sensor information.

3.2 Terms defined in this Recommendation

This Recommendation defines the following terms:

3.2.1 event packet: Gesture description information, or control commands related to a gesture, or even to a gesture image.

3.2.2 gesture: A shape and/or movement generated by a user's hand, arm, or other body parts. A gesture can be predefined.

3.2.3 raw gesture data: Non-processed gesture data directly input from a motion detect device.

3.2.4 gesture description: Information of the analysed raw gesture data.

3.2.5 gesture command: The operation instruction related to a recognizable gesture. One gesture description may be associated with different commands. For example, "raise arm" may refer to "power on" or "volume up".

3.2.6 gesture list: A textual description of a gesture, the identification of the gesture or of the gesture images.

3.2.7 gesture recognition: Analysis of raw gesture data and its translation as an understandable gesture description.

3.2.8 gesture event subscription: Action performed on a gesture recognition device that allows the list of gestures or gesture commands that are received by the target gesture-controlled IPTV terminal device (TD) to be recognized and accepted.

3.2.9 gesture-recognition device: A device which is equipped with a gesture capturing sensor so that it can recognise a user's gesture and send event packet(s) to a gesture-controlled device according to the gesture-controlled device's subscription. A gesture-recognition device has capabilities such as networking, computing, storage, etc.

3.2.10 gesture-controlled device: A device which is intended to be controlled by gesture control commands.

3.2.11 gesture capturing sensor: A device that has gesture data processing ability, such as gesture data/image compression, gesture data/image manipulation with raw data captured by a somatosensory/video sensor.

3.2.12 gesture capturing sensor information: Extracted and derived information from raw gesture data, for example, shapes, outlines, directions, duration, or movement.

3.2.13 gesture-controlled device discovery: Function enabling a gesture-recognition device to discover and identify one or several gesture-controlled devices using computing identification information such as IP address and device ID, and visual features such as QR code or optical signal detected by a visual computation mechanism..

3.2.14 visual feature: A label which can be recognised with a visual computing mechanism. For example, a device's appearance picture, bar code, quick response (QR) code or optical signal, etc.

3.2.15 recognizable gesture: A gesture which is able to be identified and processed by a gesture recognition device or software, depending on the recognition algorithm, filter conditions and other predefined recognition principles.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

API	Application Programming Interface
EPG	Electronic Programme Guide
EUIF	Enhanced User Interface Functions
DNS	Domain Name System
IPTV	Internet Protocol TV
mDNS	Multicast DNS
QR	Quick Response
RFID	Radio Frequency Identification
SDK	Software Development Kit
STB	Set-Top-Box
TD	Terminal Device
UI	User Interface
UPnP	Universal Plug and Play

XML Extensible Markup Language

5 Conventions

The following conventions are used in this Recommendation:

– The keywords "is required to" indicate a requirement which must be strictly followed and from which no deviation is permitted, if conformance to this Recommendation is to be claimed.

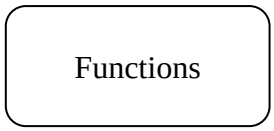
– The keywords "is prohibited from" indicate a requirement which must be strictly followed and from which no deviation is permitted, if conformance to this Recommendation is to be claimed.

– The keywords "is recommended" indicate a requirement which is recommended but which is not absolutely required. Thus, this requirement need not be present to claim conformance.

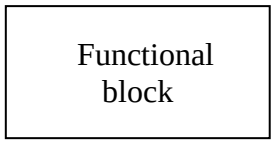
– The keywords "is not recommended" indicate a requirement which is not recommended but which is not specifically prohibited. Thus, conformance with this Recommendation can still be claimed even if this requirement is present.

– The keywords "can optionally" indicate an optional requirement which is permissible, without implying any sense of being recommended. This term is not intended to imply that the vendor's implementation must provide the option and the feature can be optionally enabled by the network operator/service provider. Rather, it means the vendor may optionally provide the feature and still claim conformance with this Recommendation.

– The keyword "functions" is defined as a collection of functionalities. It is represented by the following symbol in the context of IPTV architecture:



– The keyword "functional block" is defined as a group of functionalities that have not been further subdivided at the level of detail described in this Recommendation. It is represented by the following symbol in the context of IPTV architecture:



6 Introduction

6.1 The concept enhanced user interface functions for gesture control

A gesture control interface used in an IPTV service helps an end-user to control and interact with the IPTV terminal device without the use of a conventional remote controller. Users can easily change the IPTV application status, e.g., channel change, volume up and down, according to their gestures or special body motions. Currently, many terminal devices have been embedded with a dedicated gesture/motion detect device, such as a gyroscope, G-sensor, etc. Such equipment could help users create a gesture with a control intention. However, in this case the gesture detection, gesture recognition and gesture-controlled functions are merged in one IPTV terminal device (TD). The IPTV TD is only interacting with one user. In the current market, the technology of gesture

detection and gesture data accessing are provided by each manufacturer's software development kit (SDK). Hence, the above case is out of scope of this Recommendation.

In the other cases, the gesture recognition device is a shareable device, which could be a dedicated device, or a software component embedded in an IPTV TD. It can provide gesture recognition service for many terminal devices. So, the gesture enhanced UI functions (EUIF) in each controllable IPTV terminal device can help to discover an available gesture recognition device and subscribe to the supported gesture events notification. In addition, the discovery of the gesture controllable device and gesture recognition device can be bi-directional. The gesture EUIF may provide a user interface for a user to configure its gesture profile, such as adding a gesture, delete a gesture, updated gesture event lists.

By considering the background case mentioned above, in this Recommendation, a new specification of gesture EUIF is introduced based on the framework of [ITU-T H.703]. The fundamental functions are also based on [ITU-T H.703], as shown in Figure 6-1. It shows the basic concept of enhanced UI functions in IPTV end user functions.

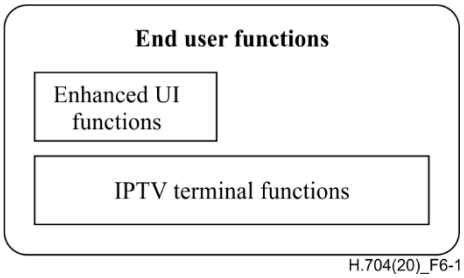


Figure 6-1 – Enhanced UI functions in case of IPTV terminal device

Figure adapted from [ITU-T H.703].

In [ITU-T H.703], the EUIF of touch screen and speech interface were introduced. In this Recommendation, the EUIF can be extended to provide one more possible way of controlling IPTV TD by connecting to a gesture detection device. However, gesture control is more complex than speech interface and touch screen. It may rely on many comprehensive factors such as location, distance and gesture recognition capabilities.

The major function of gesture EUIF is to enable an IPTV TD to recognize a gesture or transfer the gesture into a command which can be supported or understood by IPTV TD. An IPTV TD enabled by gesture EUIF can be differentiated from a gesture-controlled device or a gesture recognition device, according to the different EUIF functions.

Basically, in the IPTV TD software framework, the gesture EUIF is the middle layer between IPTV application and hardware driver. It provides a common application programming interface (API) for IPTV application to call the gesture capturing sensor functions.

Moreover, the gesture EUIF should also provide a capability of understanding the control intention of an end-user for the IPTV TD, based on the captured gesture (see information on related standardization in Appendix III). For example, "arm raise" may be regarded as "volume +", "channel +", "light +", etc. Gesture EUIF receives "arm raise" and should determine which meaning is right for the IPTV TD.

In another case, if IPTV TD only supports commands such as "volume +/-" and "channel +/-", gesture EUIF should not receive any other gesture events except those two options.

If a mapping of a gesture-command is used in a gesture recognition device, IPTV TD may receive the direct control command from the gesture recognition device.

Figure 6-2 shows the high-level procedure on how the gesture is captured and translated into a control command by utilizing a gesture control interface embedded in a gesture-controlled device and gesture recognition device.

The fundament of the gesture control for IPTV TD is the recognition of the recognizable gesture, body image, motion track or other reference object which is captured from a motion detection or gesture capturing device.

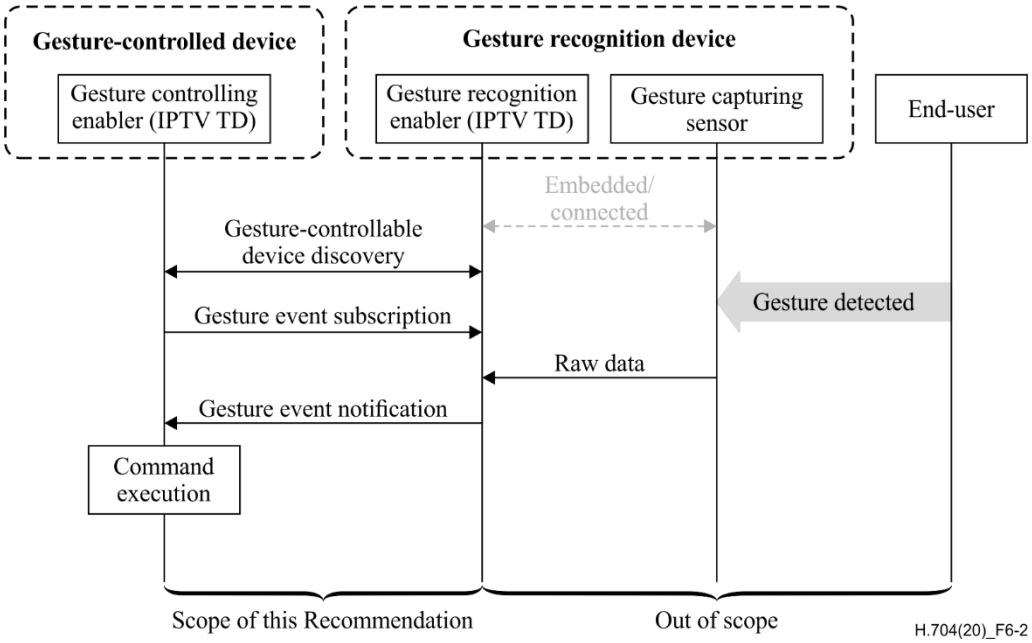


Figure 6-2 – High-level procedure of using gesture control with IPTV TD

6.2 Functional block overview

A gesture UI functional blocks, as shown in Figure 6-3, is basically composed of a gesture controlling enabler and a gesture recognition enabler as follows:

– **Gesture control functional block**: supports gesture control functionality between user and IPTV terminal device

– **Gesture controlling enabler**: enables a terminal device to be controlled by a gesture recognition device

– **Gesture recognition enabler**: provides the gesture recognition functionalities that enable the capturing of gestures

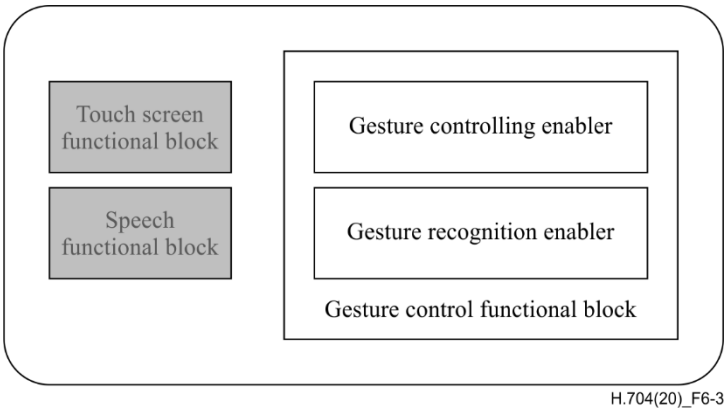


Figure 6-3 – Logical functional block diagram

7 Requirements

General UI requirements for IPTV service are defined in clause 7 of [ITU-T H.703]. This clause presents specific requirements for gesture recognition and control.

7.1 Gesture control functional block requirements

The following are requirements regarding gesture control functions in an IPTV terminal device:

– Gesture control function is required to provide end-user interaction based on gesture commands.

– Gesture control function is recommended to support gesture commands mapped to an original IPTV service control command.

– Gesture control function is recommended to discover one or more gesture capturing sensors locally or remotely.

– Gesture control function is recommended to support connection to a gesture capturing sensor by at least one of the following models:

  • Local embedded camera device

  • Direct USB connection

  • Network connection

– Gesture control function is recommended to support network functions.

– Gesture control function is able to store the IPTV TD location, device ID, network address and other identification information.

7.2 Gesture controlling enabler requirements

The following items are requirements regarding gesture controlling enablers in IPTV terminal device:

– The gesture controlling enabler is recommended to support the mapping between a gesture and IPTV control commands.

– The gesture controlling enabler is recommended to support the mapping of gestures to other IPTV TD commands such as launching applications, displaying electronic programme guide (EPG) or system menus.

– The gesture controlling enabler is recommended to support user's previous registry of which gestures will be mapped to which commands.

– The gesture controlling enabler is recommended to store the list of supported gestures or mapped control commands.

– The gesture controlling enabler is recommended to support the discovery and connection to a gesture recognition device.

– The gesture controlling enabler is recommended to support exposure of the device's computer vision feature, for example, the device's appearance image, QR code, beam signals, etc.

– The gesture controlling enabler is recommended to publish the list of recognizable gestures or mapped commands in a network.

– The gesture controlling enabler is recommended to receive the gesture description or mapped command from a gesture recognition device.

7.3 Gesture recognition enabler requirements

The following are requirements regarding gesture recognition enabler:

– The gesture recognition enabler is recommended to detect and capture the raw gesture data.

– The gesture recognition enabler is recommended to identify a gesture-controlled device, including device location, device ID, network address, etc.
– The gesture recognition enabler is recommended to recognise a gesture and the target gesture-controlled device at the same time.
– The gesture recognition enabler is recommended to send the gesture description or mapped command to a target gesture-controlled device.
– The gesture recognition enabler is recommended to interact with other gesture recognition enablers.
– The gesture recognition enabler may be embedded in the gesture capture device, i.e., the gesture-recognition device.

8 Gesture UI functionality and interfaces

8.1 Gesture control functional block

In general, the gesture control functional block provides the gesture input interface for an IPTV user. It captures the user's gesture and analyses it. In addition, the gesture control functional block also provides the identifying of the gesture-controlled devices, i.e., to identify the location of the gesture-controlled device, the device ID, the network ID (network address) and the published gesture list supported by the gesture-controlled device. Moreover, once a gesture from a user is recognized, the gesture control functional block will send that gesture description to the device which the user intends to control.

The gesture control functional block is recommended to support the mapping of gesture commands to the original IPTV service control commands.

The gesture control functional block, logically, is composed of a gesture-controlled functional component and a gesture recognition functional component. But physically, the gesture-controlled functional component is usually embedded into a gesture-controlled device. On the other hand, the gesture recognition functional component is embedded into a gesture-recognition device.

The detailed functionality of the gesture-controlled functional component and the gesture-recognition functional component are described in clauses 8.1.1 and 8.1.2.

8.1.1 Gesture controlling enabler

The gesture controlling enabler enables the IPTV terminal device to be controlled following a gesture instruction. It may be composed of several functionalities, as shown in Figure 8-1.

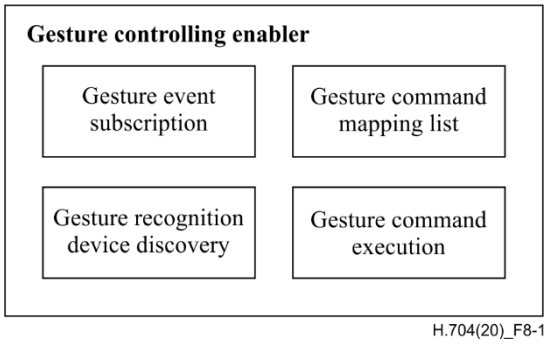


Figure 8-1 – Functionalities in gesture controlling enabler

Gesture recognition device discovery

The gesture-controlled device should be discoverable and identifiable by sending its identification information to a gesture-recognition device. The identification information includes two types of elements: the computing identification and the visual features.

The computing identification refers to the parameters that can be used in networking messages, such as IP address, device ID, user ID, etc.

The visual feature refers to the appearance or the profile which can be detected by a visual computation mechanism. For example, a device's appearance picture, QR code, optical signal, etc.

With those features, the gesture-controlled device can be discovered, identified and connected to a gesture-recognition device by device interworking protocols, such as UPnP [b-ISO UPnP] or mDNS [b-IETF RFC 6762].

Gesture command mapping list

The gesture command mapping list maintains a list of textual gesture descriptions and the respective mapping to a command. It is noted that one textual description may be mapped to several commands. For exampe: "raise arm" may refer to "power on" or "volume up". For special cases, the gesture description may be mapped to a null command without means. For example: "waving palm" could refer to "power off" for some set-top-boxes (STBs) but means "null" in the other STBs.

IPTV service command can be configured in the list by mapping the gesture to an IPTV service control command.

It is noted that the gesture command mapping list can be sent together with the gesture event subscription request alternatively. That list can be used to create the gesture event subscription list.

An example of a gesture command mapping list can be referenced from Table 1.

It is noted that Table 1 is an example of a gesture command mapping list. Many other gesture descriptions can be added into this list by dynamically configuring from user's expectation.

Table 1 – Example of gesture command mapping list

Gesture command mapping list		
Gesture description	**Additional information**	**Command**
"raise arms over shoulder"	"coordination"	Power on/off
"turn around"		null
"raise the left hand"	"duration"	Volume +
"put down the left hand"	"duration"	Volume -
"raise the right hand"	"times"	Channel +
"put down the right hand"	"times"	Channel -
……		……

Gesture event subscription

The gesture-controlled device needs to publish its supported gesture command mapping list into network when it connects to a gesture-recognition device. Once the subscription is succeeded, it can receive the gesture event packets from the gesture-recognition device.

Besides the control command, there are two types of gesture descriptions in the gesture event packet that can be received: the default gesture description, which contains only the gesture name, ID, or other single identification; the other one is the detailed gesture description, which contains the additional parameters such as the coordination of gesture in the space, duration, speed, etc.

Gesture command execution

As shown in Figure 8-2, once a gesture event packet is received, the gesture command execution needs to extract the information from the gesture event packet and then select the related operations:

– If the information is a control command, which is an explicit IPTV command, the operation corresponding to that command can be activated.

– If the information is a gesture description, the gesture command execution needs to firstly analyse the gesture description by working with the gesture command mapping list. If there is an IPTV command mapped to that gesture description, the operation corresponding to that command can be activated.

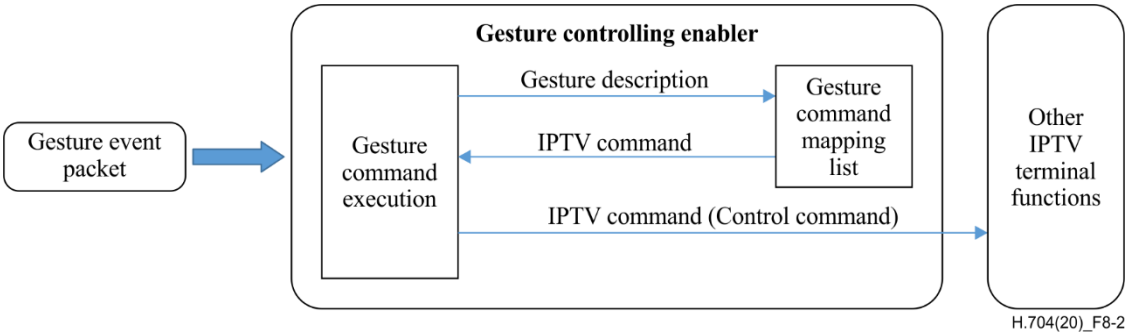


Figure 8-2 – Example process of gesture event packet

With the above functionalities, once the controlled device receives a gesture event packet from the recognition device, it should execute the related operations according to the predefined logical functions.

8.1.2 Gesture recognition enabler

The gesture recognition enabler provides the capture of a raw gesture and its interpretation. It also provides the gesture description to the gesture-controlled device. For the case of multiple devices controlling, the gesture recognition enabler needs to identify the available gesture-controlled device(s) within its visual range before gesture capture and recognition. It may be composed of the functionalities shown in Figure 8-3.

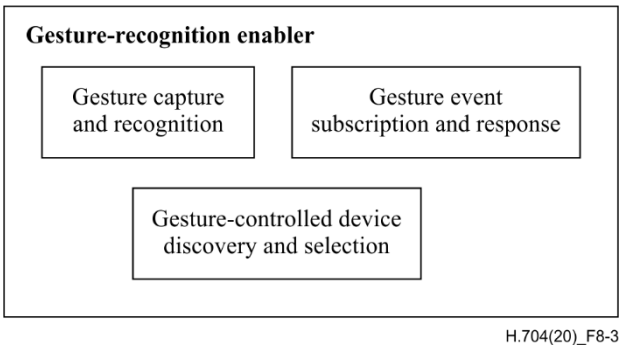


Figure 8-3 – Functionalities in gesture recognition enabler

Gesture capture and recognition

This functionality enables the recognition device to detect and capture a raw gesture data from a user presentation and to translate the raw gesture data into a textual gesture description. In addition, this functionality can be invoked by gesture-controlled device discovery functions for identifying a controllable terminal device. Once the gesture capture and recognition are completed, a gesture event packet will be created by wrapping the gesture description or control command.

Gesture-controlled device discovery and selection

The gesture-controlled device discovery enables the recognition device to discover and identify one or several controlled devices with the device identification information mentioned in clause 8.1.1.

In the visual discovery and identification process, the recognition device retrieves the visual features from messages sent by the controlled devices. To identify the target-controlled device, the recognition device should capture the controlled devices visual features by a gesture capturing sensor and then compare it with the previously received visual features.

As a result, the address, device ID and position of the gesture-controlled device will be stored for the gesture event subscription and response. In addition, the position data can be updated periodically.

To select a candidate-controlled device from multiple devices, the recognition device is recommended to not only recognize the gesture (raw gesture data), but also recognize the target-controlled device to which the gesture directs.

In order to identify the target-controlled device mentioned above, it is recommended to calculate the gesture direction by using an angle between the gesture and the gesture-controlled device.

The alternative way is to measure the distances between the gesture, the controlled device and the gesture recognition device so that the gesture direction can be calculated by using trigonometric formulas.

The visual identification process is invoked before gesture capture and recognition. The controlled devices' computing identification and location are stored in the recognition device. It is noted that other technologies besides visual discovery may be used to assist the discovery process where multiple controllable devices are present, for example radio frequency identification (RFID) and beacons. These other technologies may not only assist on device discovery, but also on device positioning inference.

Gesture event subscription and response

The gesture event subscription and response function receives the subscription request from the controlled devices. When a gesture event is detected, it is able to transmit the gesture event packet to a controlled device according to the gesture command mapping list sent by the controlled device previously.

If there are more than one candidate of controlled devices, the target device needs to be selected before the event transmission. A gesture event subscription list can be used to help to find the target-controlled device and to select the right control command.

Table 2 shows an example of a gesture event subscription list.

Table 2 – Example of gesture event subscription list

Gesture event subscription list				
Gesture description	IPTV TD1		IPTV TD2	IPTV TD3
	Mapping list		Mapping list	No mapping list
"raise arms over shoulder"	Power on/off			N/A
"raise the left hand"	Volume +	"Twice"	brightness +	N/A
"put down the left hand"	Volume -		brightness -	N/A
"raise the right hand"	Channel +	"3 seconds"	"null" or n/a	N/A
"put down the right hand"	Channel -			N/A

After the recognition device recognizes a gesture which directs to a controlled device, it transmits the gesture event packet to the target-controlled device. The gesture event packet includes the gesture description or the control command corresponding to the gesture command mapping list. Therefore, the controlled device can implement the related operations according to the received gesture event packet. Figure 8-4 shows an example of elements in gesture event packet.

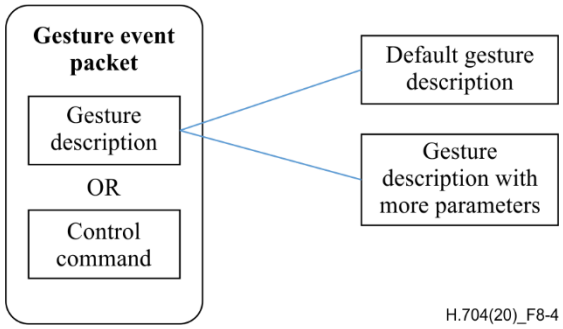


Figure 8-4 – Example of elements in gesture event packet

It is noted that, in IPTV service, it is possible for one recognition device to interact with one or more controlled devices at the same time.

8.2 Interfaces between functional components

According to the scenarios described in Appendix II, there are four cases of deploying gesture control functional components in the IPTV terminal device. However, there are three logical interfaces recommended to be always implemented.

The logical interfaces are shown on the Figure 8-5.

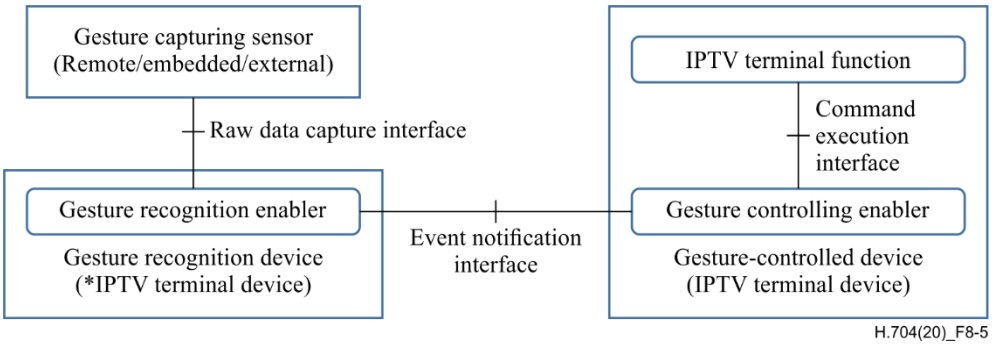


Figure 8-5 – Interfaces between gesture control functional components

NOTE – Gesture recognition device could be deployed in the same IPTV terminal device with a gesture controlling enabler. However, this Recommendation is only focused on the case of the deploying a gesture recognition device and gesture-controlled device in the different IPTV terminal devices. The gesture recognition device could be a dedicated device as well.

8.2.1 Raw data capture interface

This interface is between the gesture capturing sensor and gesture recognition enabler. It is used to deliver the raw gesture data to a gesture recognition enabler. If the gesture capturing sensor is physically separated from the gesture recognition device, this interface is recommended to be implemented as network connection. If the gesture capturing sensor is an external equipment or embedded in the gesture recognition device, this interface is recommended to be implemented as an internal API.

8.2.2 Event notification interface

This interface is between the gesture recognition enabler and gesture controlling enabler. It is used to exchange the gesture event data between those two enablers.

The gesture event data is composed of three elements (reused event data elements defined in [ITU-T H.703]), as shown in Table 3:

Table 3 – Elements for gesture event data

Element	Description
Event type	This element indicates the type of the gesture event. The gesture events can be divided into two types: Gesture event subscription and Gesture event notification: For example, the event type can be expressed as follows: – EV_SUB: indicates that the event is a request of gesture event subscription from Gesture controlling device. – EV_NTF: indicates that the event is a notification of gesture event to the selected Gesture-controlled device when the gesture event packet is available.
Event code	When the event type indicates the gesture event subscription, this element denotes the gesture description with/without additional information, e.g., a gesture-command mapping list, which the gesture-controlled device needs to receive. For example, the event code can be expressed as follows: – SUB_G_Desc: contains the code/textual description of a gesture. – SUB_G_C_Map: contains an XML/table of gesture-command mapping list.
	When the event type indicates the gesture event notification, this element denotes the gesture description or control command with/without additional information that the gesture-controlled device needs to receive. For example, the event code can be expressed as follows: – NTF_G_Desc: contains the textual description/code of a gesture. – NTF_Ctrl: contains an explicit IPTV control command/code.
Value	When the event code indicates the gesture description, this element denotes the text of the gesture description. For example: – "left hand up", "0x0002", "twice"
	When the event code indicates the gesture-command mapping list, this element denotes a list file or a table. For example: – **Table 1** defined in 8.1.1
	When the event code indicates the control, this element denotes the text of the IPTV command. For example: – "channel +", "0x0013"

8.2.3 Command execution interface

This interface is between the gesture controlling enabler and the IPTV terminal function. It is used to implement the operation to the IPTV service, according to the end-user's intention. If it is possible, this interface also can be used for the gesture controlling enabler to access the IPTV terminal function to acquire the code of IPTV command.

9 High-level procedure flows

In this clause, a set of high-level procedure flows are presented. In this Recommendation, as an instance, it is assumed that the IPTV terminal device is embedded with the gesture controlling enabler and acts as a gesture-controlled device. The gesture recognition device could be an IPTV TD with a camera or other types of device.

9.1 Procedure of gesture control enabled device discovery

This clause describes the procedure of gesture control enabled device discovery. There are two scenarios: gesture recognition device-initiated device discovery and gesture-controlled device initiated discovery.

9.1.1 Procedure of gesture recognition device-initiated discovery

In this scenario, it is assumed that the gesture recognition device is first online or rebooted. Figure 9-1 shows the procedure of device discovery.

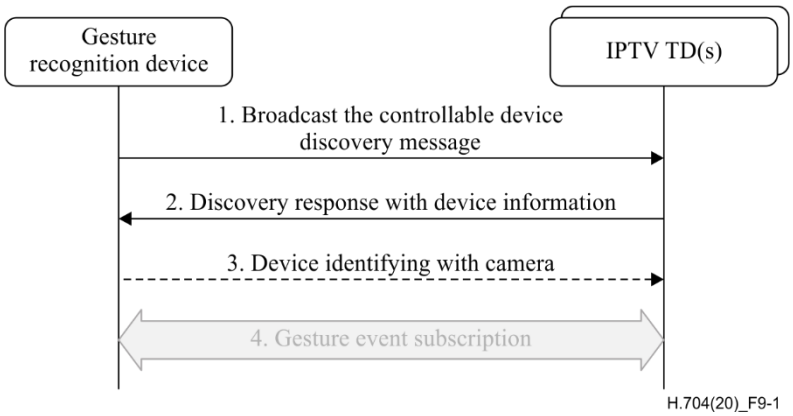


Figure 9-1 – Procedure of gesture recognition device-initiated discovery

In this procedure:

1) When the gesture recognition device is available, it broadcasts a gesture-controlled device discovery message in a local network. In the discovery message, the information about the gesture recognition device may be included, such as the IP address, functional roles description, etc. The discovery message can be optionally broadcasted periodically.
2) The IPTV TDs, which received the discovery message will choose to feedback the discovery response message. In the response message, the information about the IPTV TD will be attached, such as Device ID, IP address, position, etc.
3) According to the received response, the gesture recognition device starts the device detecting and identifying within its visual range by using a video capture device, e.g., a camera. The information of a device that is out of the visual range may be abandoned.
4) Once the target gesture-controlled devices are identified, the gesture recognition device can communicate with them by receiving the gesture event subscription requests.

It is noted that, once the gesture recognition device is available, it will keep listening to the information sent from the discoverable gesture-controlled devices subsequently.

9.1.2 Procedure of gesture-controlled device-initiated discovery

In this scenario, it is assumed that a gesture-controlled device is newly added into the local network or rebooted. Figure 9-2 shows the procedure of gesture-controlled device-initiated discovery.

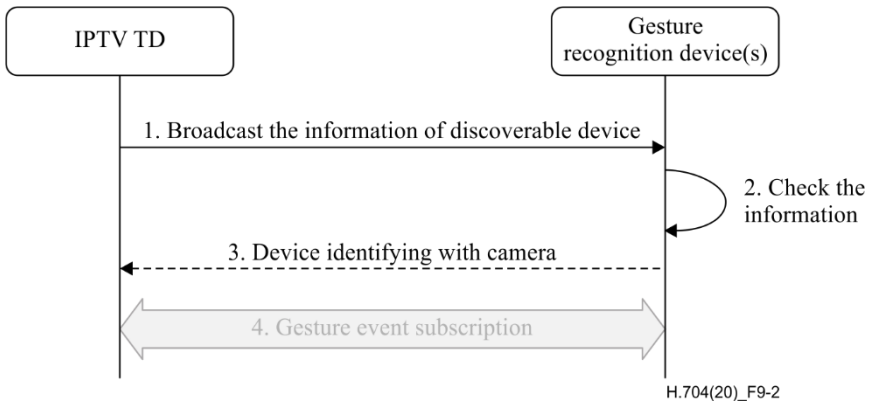


Figure 9-2 – Procedure of gesture-controlled device-initiated discovery

In this procedure:

1) When the IPTV TD is available, it broadcasts a discoverable message of IPTV TD in a local network. The message includes its identification information, such as the IP address, device ID, etc.

2) The gesture recognition device (maybe more than one), will check the discoverable message. If the information of IPTV TD has been received previously, it only needs to update its subscription list. If the information is a new one, it will start the implementation of Step 3.

3) According to the received discoverable message, the gesture recognition device starts the device detecting and identifying within its visual range by a video capture device, e.g., a camera. The information of a device that is out of the visual range may be abandoned.

4) Once the target gesture-controlled device is identified, the gesture recognition device can communicate with it by receiving the gesture event subscription requests.

It is noted that, one IPTV TD can be subscribed by multiple gesture recognition devices. However in this Recommendation, as in the case study, it is appropriate to assume that there is only one gesture recognition device in the local network.

9.2 Procedure of gesture event subscription

Figure 9-3 shows the procedure of gesture event subscription.

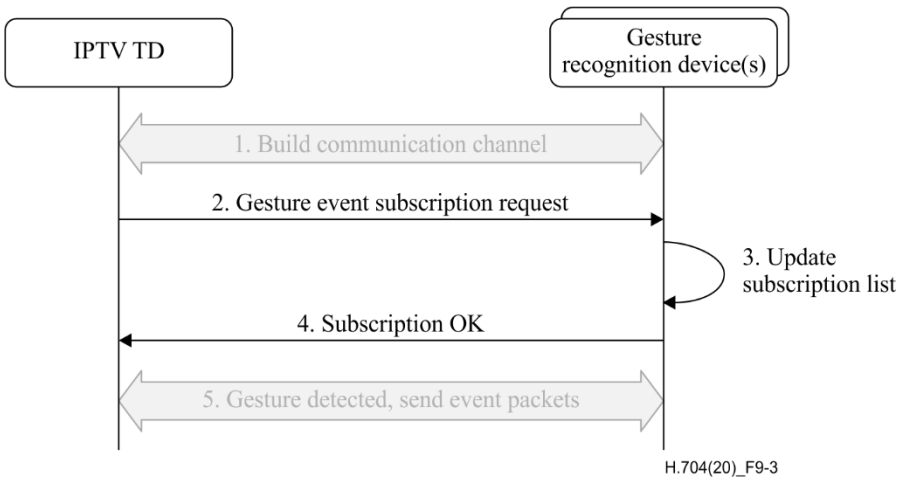


Figure 9-3 – Procedure of gesture event subscription

Procedure description:

1) Once the IPTV TD is recognised by the gesture recognition device, a communication channel between them is built for exchanging the future event data. The procedure of device interconnection can be referenced from [b-ITU-T H.772].
2) The IPTV TD send the gesture event subscription request to the gesture recognition device. A gesture command mapping list is attached into the request. If there are multiple gesture recognition devices, the gesture event subscription request may be published to all of them simultaneously.
3) The gesture recognition device updates its subscribed device list with the new parameter contained in the subscription request.
4) If the subscription succeeds, the gesture recognition device will send back a conformation message.
5) Once a user gesture is detected, the gesture recognition device will send the gesture event packet to all the gesture-controlled devices in the subscribed device list.

It is noted that, if an IPTV TD is subscribed by multiple gesture recognition devices, it may not receive the same gesture event from the gesture recognition devices which do not support the detection of a specific gesture.

9.3 Procedure of user gesture event recognition

Figure 9-4 shows the procedure of user gesture event recognition.

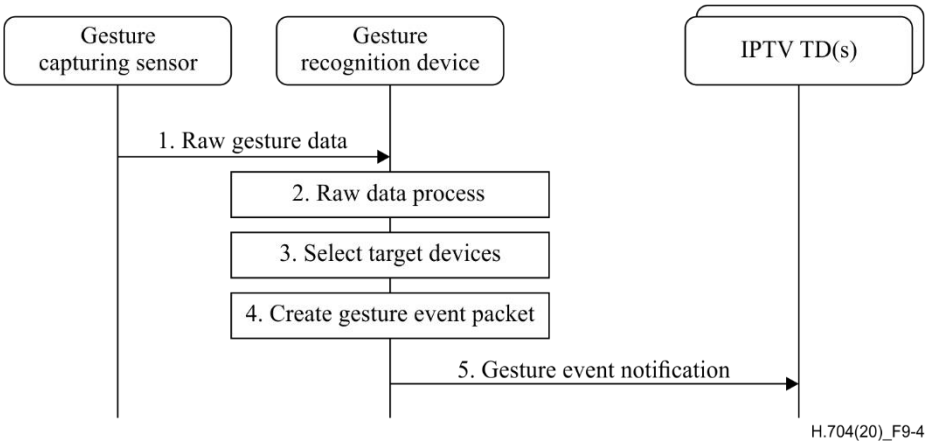


Figure 9-4 – Procedure of user gesture event recognition

In this procedure:

1) A raw gesture data is received by one or more motion sensors (e.g., a remote camera or an embedded camera). Then the raw gesture data is delivered to the gesture recognition device.
2) The raw gesture data is interpreted as the textual gesture description according to the pre-configuration. If the gesture recognition device is not able to recognize the raw gesture data, this gesture event may be ignored.
3) The gesture recognition device would select the target IPTV TD(s) by checking the event subscription (device) list. If the gesture description contains additional information, such as the coordinates of gesture in the space, angle, etc., it may be easy for the gesture recognition device to select a specific IPTV TD. If there is no such event subscription, the gesture description may be dropped.

4) Once the target-controlled IPTV TD is selected, a gesture event packet would be created by attaching the gesture description information into it. Moreover, by checking the gesture command mapping list, the gesture recognition device may replace the gesture description with a specific IPTV control command.

5) The gesture recognition device would notify the controlled device(s) of a new gesture event by sending the gesture event packet. In addition, the related reaction of the gesture event may not be completed or stopped until the next gesture event is received.

It is noted that, logically, a gesture recognition device is able to control multiple IPTV TDs simultaneously. However, for the best user control experience, it is recommended to control the IPTV TDs in an order if there are multiple candidate TDs.

Appendix I

Examples of descriptions related to the gesture-enabled service

(This appendix does not form an integral part of this Recommendation.)

The following two clauses show examples to describe the gesture controlling device – STB and the related gesture-enabled service.

I.1 Example of an XML description for a gesture controlling device – STB

```
<device>
      <deviceType>urn:schemas-itu:device:gesture_recognition:1</deviceType>
      <friendlyName>Example_STB_Device</friendlyName>
      <manufacturer>EXAMPLE</manufacturer>
      <manufacturerURL>http://www.example.com.cn</manufacturerURL>
      <modelDescription>EXAMPLE_STB_Device</modelDescription>
      <modelName>Set-top-box</modelName>
      <modelNumber>1.0</modelNumber>
      <modelURL>http://www.example.com.cn</modelURL>
      <serialNumber>0000000003</serialNumber>
      <UDN>uuid:20140315-f300-f400-f501-a1234567890a</UDN>
      <UPC>123456789013</UPC>
      <iconList>
            <icon>
                  <mimetype>image/gif</mimetype>
                  <width>48</width>
                  <height>32</height>
                  <depth>8</depth>
                  <url>icon.gif</url>
            </icon>
      </iconList>
      <serviceList>
            <service>
                  <serviceType>urn:schemas-itu:service:gesture:1</serviceType>
                  <serviceId>urn:schemas-itu:serviceId:gesture1</serviceId>
                  <SCPDURL>/service/gesture/service.xml</SCPDURL>
                  <controlURL>/service/gesture/control</controlURL>
                  <eventSubURL>/service/gesture/eventSub</eventSubURL>
            </service>
      </serviceList>
      <presentationURL>http://www.example.com.cn</presentationURL>
</device>
```

I.2 Example of an XML description for a gesture-enabled service within a gesture controlling device

```
<actionList>
      <action>
            <name>GestureList</name>
            <argumentList>
                  <argument>
                        <name>device_name</name>   <!—the name of controlled device-->
                        <relatedStateVariable>default_string</relatedStateVariable>
                        <direction>out</direction>
                  </argument>
                  <argument>
                        <name>gesture_name</name>
                        <relatedStateVariable>default_string</relatedStateVariable>
                        <direction>out</direction>
                  </argument>
                  <argument>
                        <name>gesture_of_action_name</name>
                        <relatedStateVariable>default_string</relatedStateVariable>
                        <direction>out</direction>
                  </argument>
```

```
            </argumentList>
        </action>
        <action>
            <name>GestureResult</name>
            <argumentList>
                <argument>
                    <name>device_name</name>     <!—the name of recognition device-->
                    <relatedStateVariable>get_string</relatedStateVariable>
                    <direction>in</direction>
                </argument>
                <argument>
                    <name>action_name</name>
                    <relatedStateVariable>get_string</relatedStateVariable>
                    <direction>in</direction>
                </argument>
                <argument>
                    <name>gesture_parameters</name>
                    <relatedStateVariable>get_string</relatedStateVariable>
                    <direction>in</direction>
                </argument>
            </argumentList>
        </action>
    </actionList>
    <serviceStateTable>
        <stateVariable sendEvents="no">
            <name>default_string</name>
            <dataType>string</dataType>
        </stateVariable>
        <stateVariable sendEvents="no">
            <name>get_string</name>
            <dataType>string</dataType>
        </stateVariable>
    </serviceStateTable>
```

Appendix II

Four scenarios where an IPTV TD can connect to a gesture capturing sensor

(This appendix does not form an integral part of this Recommendation.)

There are four possible scenarios for an IPTV TD to connect to a gesture capturing sensor (e.g., a camera), described in the following clauses.

II.1 IPTV TD with embedded gesture capturing sensor

In this case, IPTV TD has a built-in gesture capturing sensor and it can be driven directly by the IPTV TD application. The camera could be used for motion control if a special motion recognition application is designed. EUIF could be the software framework for installing a motion recognition application.

In addition, in this situation, IPTV TD can provide the capability of gesture subscribing and packet event for other IPTV terminal devices.

II.2 IPTV TD with external gesture capturing sensor

In this case, IPTV TD can connect to a gesture capturing sensor with the external I/O port. For example, a user can plug a webcam to an IPTV TD with a USB port. The usage of an external gesture capturing sensor is very similar to the case presented in clause II.1. It is noted that, both embedded gesture capturing sensors and external gesture capturing sensors should have a driver program installed into the IPTV TD driver library.

II.3 IPTV TD with remote gesture capturing sensor

In this case, IPTV TD does not have any embedded or external gesture capturing sensor but can connect to a remote gesture capturing sensor with a network connection. The IPTV TD can receive the motion signal from the remote gesture capturing sensor and transfers them into an IPTV control command by EUIF software.

The cases presented in II.1, II.2 and II.3 are all required to enable the gesture recognition function in the IPTV TD, which means the IPTV terminal is a combination of gesture-controlled device and gesture recognition device.

II.4 IPTV TD with remote gesture capturing sensor combined with recognition device

In this case, IPTV TD is connected to a gesture capturing sensor combined with a recognition device. The recognition device processes the captured image from the gesture capturing sensor and transfers it into the control command. The command is sent to IPTV TD to implement a control option.

Under this situation, after the building of connection between an IPTV TD and a gesture capturing sensor, the IPTV TD could send the gesture subscribing to that gesture capturing sensor and receive a packet event from it.

It is noted that, both cases II.3 and II.4 are not required to install the sensor driver in advance. However, before the IPTV TD can receive any signal from a remote gesture capturing sensor, it should communicate with the remote sensor by EUIF for harmonizing its work. For example, in case II.4, the IPTV TD should tell a remote camera what is the meaning of a gesture and what options are supported.

Appendix III

Summary of relevant standardization activity

(This appendix does not form an integral part of this Recommendation.)

In this appendix, other SDO's relevant activity is briefly summarized in the relevant specification title and key scope.

The W3C Note on Multimodal interaction framework [b-W3C MMIF] identifies the major components for multimodal systems. Refer also to http://www.w3.org/2002/mmi/.

Bibliography

[b-ITU-T H.742.0] Recommendation ITU-T H.742.0 (2016), *Use of video sensor devices for IPTV services: Architecture and requirements.*

[b-ITU-T H.772] Recommendation ITU-T H.772 (2015), *IPTV terminal device discovery*.

[b-ISO UPnP] ISO/IEC 29341-1:2011, *Information technology – UPnP Device Architecture – Part 1: UPnP Device Architecture Version 1.0*.

[b-IETF RFC 6762] IETF RFC 6762 (2013), *Multicast DNS*.

[b-W3C MMIF] W3C Note (2003), *W3C Multimodal Interaction Framework*. https://www.w3.org/TR/mmi-framework/

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	**Audiovisual and multimedia systems**
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2020