INTERNATIONAL STANDARD

ISO/IEC 30113-11

First edition
2017-08

Information technology — Gesture-based interfaces across devices and methods —

Part 11: **Single-point gestures for common system actions**

Technologies de l'information — Interfaces gestuelles entre dispositifs et méthodes —

Partie 11: Gestes à point unique pour actions courantes du système

Reference number
ISO/IEC 30113-11:2017(E)

ISO copyright office
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

Contents

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.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

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. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see the following URL: www.iso.org/iso/foreword.html.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 35, *User interfaces*.

A list of all parts in the ISO/IEC 30113 series can be found on the ISO website.

Introduction

A single-point gesture is generated by a user with a pointer [which includes an input device (e.g. a mouse, a stylus, etc.) or a body part (e.g. a fingertip, a hand, etc.)] providing gesture actions using a single point of interest (POI). The gesture can be recognized and interpreted as a command which is understood by platforms, systems or applications running information and communications technology (ICT) systems. If the user feels that the gesture is natural and it is convenient and comfortable to generate, he/she can easily utilize the gesture for the applications on the ICT systems without spending an excessive amount of time learning and practising how to use the gesture.

A single-point gesture is used in several commercially-available ICT systems including personal computers, smart televisions, smart phones and video game consoles. There is a need for an International Standard to define the single-point gesture so that a user does not get confused if a gesture is associated with a different command for another application. Potential variety and inconsistency among single-point gestures might cause a serious usability problem in using applications on ICT systems.

This document presents descriptions of single-point gestures and their corresponding gesture commands for system level functions and common functions across applications on ICT systems. The system level functions include functions for selecting, initiating, terminating, resuming and restarting applications which are handled by operating systems or platforms. The common functions across applications are also identified. These functions include navigating menus, opening and closing objects, “help”, “undo”, “redo”, and so on.

The single-point gestures will harmonize with other gestures of multiple POIs. It is expected that users can execute applications more easily and without confusion by utilizing the single-point gestures. The gestures are to be performed by the users regardless of a specific recognition technique, a certain interaction method or a device. The focus of this document is on the users and their cognitive aspects.

Information technology — Gesture-based interfaces across devices and methods —

Part 11: Single-point gestures for common system actions

1 Scope

This document defines single-point gestures for common system actions used in information and communication technology (ICT) systems. It specifies movements for clear and classified gestures recognized by the systems and applications. The single-point gestures are performed using an input device (e.g. a mouse, a stylus, etc.) or a body part (e.g. a fingertip, a hand, etc.). These single-point gestures are intended to operate in a consistent manner regardless of the system, platform, application or device.

NOTE ICT systems include, but are not limited to, digital televisions, set-top boxes, video game consoles, communication devices, Internet devices, entertainment devices and personal computers.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements 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 30113-1:2015, *Information technology — User interface — Gesture-based interfaces across devices and methods — Part 1: Framework*

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— IEC Electropedia: available at http://www.electropedia.org/

— ISO Online browsing platform: available at http://www.iso.org/obp

3.1
point of interest
POI
specific point location that a user utilizes to formulate a gesture

3.2
single-point gesture
sequence of actions of moving a *POI* (3.1) by a user which are clearly recognized by the system

3.3
multi-point gesture
sequence of actions of moving multiple *POIs* (3.1) by a user which are clearly recognized by the system

EXAMPLE A gesture with two hands or a gesture with multiple fingers.

4 General

A user interface conforming to this document shall be able to recognize single-point gestures within a pre-defined space where a user performs the gestures. Once a single-point gesture is recognized by a user interface, its corresponding gesture command shall be invoked and executed.

The gestures in this document are discussed from a user's viewpoint. The gestures are defined regardless of recognition techniques, interaction methods, input devices and so on. This document emphasizes and focuses on users who perform the gestures.

The accessibility of gestures is important to meet the needs of people with disabilities and elderly people.

EXAMPLE Gestural interactions can make use of eye-tracking and/or recognition of facial expressions for some physically disabled users.

5 Types of gestures

5.1 Mediated gesture

Mediated gestures include a sequence of movements with a single-point input device which is physically connected to and provides input data to a system such as a mouse, a joystick, a track ball, etc. In order to distinguish a gesture input from other movements of the input device, such as a mouse pointing, it is important for a user to know how to activate the gesture input.

EXAMPLE The basic actions of the mediated gestures using a mouse include "clicking", "double-clicking", "dragging", and so on. In case of a mouse with two buttons, its right button can be used for activating a gesture. Then, the gesture enters the initial state.

In order to minimize errors of misinterpreting a gesture input, a system ignores any unintentional movement of a pointing device in generating gestures. For a mediated gesture using a mouse, the minimum movement of the pointer of the mouse shall be over a specific number of pixels (e.g. 30 pixels) along horizontal, vertical or diagonal directions.

After activating a gesture input, the system enters into intermediate states. A visual trail (which presents pointer's movements and guides) should be displayed on an output device, such as a screen, in order to provide a feedback and/or a feed-forward to a user. This can improve a usability of the system or the platform. However, in case of a gesture command that contains continuous manipulation of a POI (e.g. in case of the "Displace" gesture), no visual trail should be displayed because it may interfere with a pointer or a cursor.

5.2 Direct touch gesture

A direct touch gesture is a sequence of movements (e.g. pinching, swiping, etc.) of a user with a body part (e.g. a finger) or a physical object (e.g. a stylus) on an input device (e.g. a touch pad or a touch screen). As soon as a user touches the pad or the screen, an initial state of a gesture starts. Then, its intermediate states begin as the movements of the touch (e.g. swiping, pressing, holding, etc.) proceed.

There are several elementary actions of the direct touch gestures that are important to a range of the gestures.

- "Tapping": The tapping action is performed by tapping a pad or a screen with a body part or a physical object manipulated by a user. It is a gesture that can be defined as a brief contact between the user and the aimed device. It starts with a contact and is rapidly followed by a lack of contact. The input device might sense the pressure from the body part or the physical object. This action can be used as a selector of an object. An action of "double-tapping" is performed as consecutive actions of tapping. The double-tapping can be used as an activator or an initiator of an application.

- "Holding": The holding action is started with a direct contact (i.e. "Tapping") to a pad or a screen by a user. Then, the position and the area of the tapping body part or the physical object do not change over the specific time period. The difference between the holding action and the tapping action is the time of touching the pad or the screen.

- "Pressing": The pressing action is performed by a combination of the tapping action and the holding action on a pad or a screen with a body part or a physical object by a user. The input device might sense the pressure from the touch or the changes of the touching area involved in the touch.
- "Swiping": The swiping actions are similar to the four-directional gestures of Table 1. The major difference is a speed of performing the gestures. The swiping action requires a performance with a speed. The specific speed of the swiping action shall be defined in its corresponding input device.

5.3 Non-contact gesture

A non-contact gesture involves a sequence of movements of a body part or a physical object (which is physically not connected or wirelessly connected to a system) by a user to invoke a command. It is assumed that the gesture does not touch an input screen or an input surface of the system. The non-contact gesture is generally performed in a two-dimensional or three-dimensional (2D or 3D) space where the gesture should be clearly recognized by sensors or cameras of the system.

5.4 Descriptions of gestures

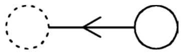
The representations of the gestures in this document shall conform to the following rules.

- If a position of a body part is involved, the gesture representation involves its starting position and ending position. A user is expected to make a clear distinction between a starting point and an ending point of each non-contact gesture for the clear recognition of the gesture.
- This document defines the starting point of the gesture as an initial state and the ending point of the gesture as a final state.
- The starting point is visually represented by a single circle with a solid line (which represents a POI as shown in Figure 1), while the ending point is represented by a circle with a dotted line as shown in Figure 2.

Figure 1 — Single circle represents a starting point of a gesture

Figure 2 — Ending point of a gesture represented by a circle with a dotted line

- An arrow is used for representing a single stroke with a specific direction. The following example presents a gesture of moving a POI from its current position (the starting point) to its left side until it arrives the ending point.



EXAMPLE 1 The "Left" gesture (G11-1 in Clause 8).

- A number of arrows represent a speed of performing gestures. The speed is expressed by adding one more arrow as shown in Figure 3. The two arrows express a "faster" stroke of the "Left" gesture. It describes a "swiping" gesture of the direct touch gesture, which is a variation of the "Left" gesture;

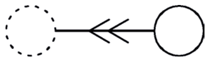
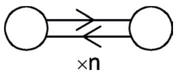


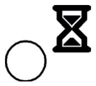
Figure 3 — Variation of the "Left" gesture, i.e. "Left swiping"

- A combination of lines and arrows are used for expressing repetitive strokes of waving. If the number of repetitive strokes is predefined to be recognized, it should be specified using “xn”, where the number, “n”, denotes an unspecified number of the repetitions, as shown below.



EXAMPLE 2 The “Horizontal wave” gesture (G11-13 in Clause 8).

- An action of holding a POI is illustrated as the starting point icon and an icon of an hourglass. The hourglass icon symbolizes a certain period of time for holding.



EXAMPLE 3 The “Tap and hold” gesture (G11-12 in Clause 8).

6 Gesture adaptation

A 3D non-contact gesture can be cognitively complex to perform, because it requires several dimensions for a user and a device to combine. Therefore, he/she often executes a 3D non-contact gesture in a two-dimensional (2D) space.

EXAMPLE When a user has to produce a 3D non-contact gesture of drawing a sphere, he/she usually produces a circle or a set of circles on a virtual 2D temporary plane.

A user naturally optimizes his/her gestures from a kinematic point of view. This is another reason why users express their 3D non-contact gestures using their body parts in a 2D plane; it costs less in terms of energy. Muscles, paths of blood flow, bone articulators and so on require optimization in terms of kinematics.

A gesture-based interface should be able to recognize and interpret the simplified and/or optimized gestures. The gesture-based interface should be adapted in order to support a user’s performances of more convenient, comfortable and natural gestures.

Experts or advanced users tend to optimize their gesture performances. They tend to move less and to execute shorter strokes. Therefore, it is important for an interface to offer and understand predefined gestures for the expert or advanced users.

Users with disabilities or users involved in simultaneous multiple tasks can be limited as they are performing gestures: some users might be able to move only a single hand, some users might be able to move their hand only but not their fingers, some users might be able to move their eyes only, and so on. Such limitations might be due to a physiological or cognitive disability or to another parallel task. An interface shall offer a way of adapting gestures (such as “eye-tracking”, “recognition of facial expressions”, and so on) so that it enables equivalent interaction opportunities as for non-disabled users.

Gestures may deliver different meanings depending on cultural and linguistic background of their users. A gesture-based interface should be able to offer a way of adapting gestures based on personal preferences, ability and cultural and linguistic adaptability.

7 Movement metaphor

7.1 General

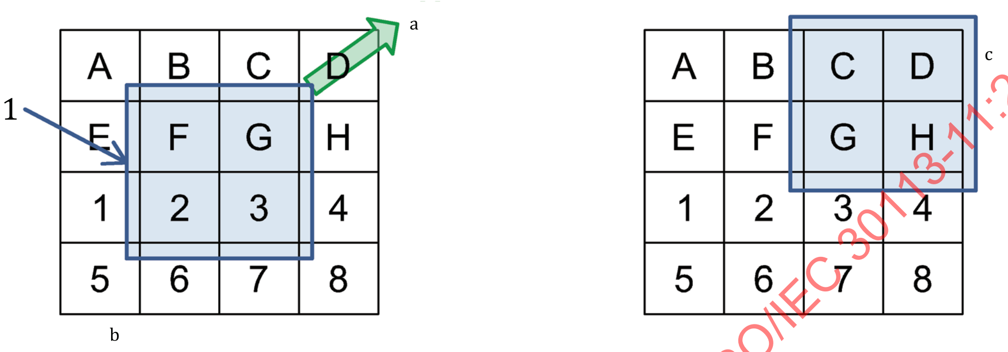
A user views digital content such as a document or an image through a viewing window of a screen on an ICT system. When the size of the viewing window is smaller than the size of the content, either the window or the content should be moved. There are two metaphors for moving a POI for viewing the content:

- moving the viewing window over the content;

— moving the content with the fixed viewing window.

The first metaphor is “movement of a viewing window”, which allows the user to navigate the content by moving the viewing window.

As illustrated in Figure 4, the user displaces the viewing window at a specific position to view a part of the content by moving a POI. In Figure 4, the user views the parts of the content, “F”, “G”, “2”, and “3” in the viewing window. If the user moves the viewing window to a new position along the direction, the user will view the parts of the content, “C”, “D”, “G”, and “H”.



Key

1 viewing window

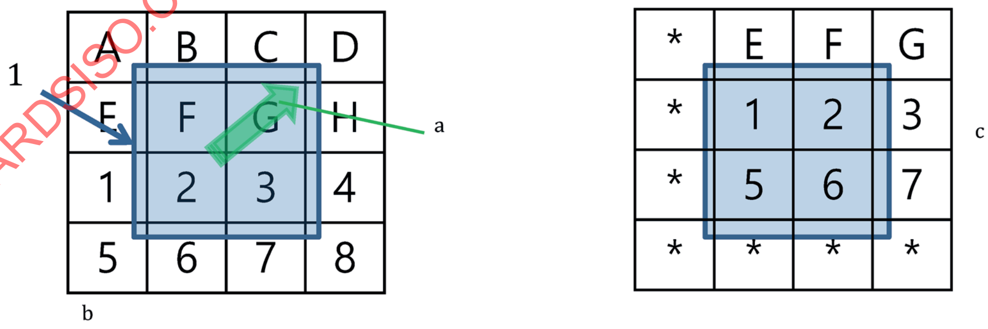
a Moving the viewing window by the “move” command from a user.

b Content to be viewed.

c Viewing window in a new position.

Figure 4 — Movement of a viewing window

The second metaphor is “movement of content”, which allows the user to navigate the content by moving a POI. Figure 5 illustrates moving the content by the user. If the user initially views “F”, “G”, “2”, and “3” through a viewing window, the user will view “1”, “2”, “5”, and “6” after moving a POI along the direction under the “movement of content” metaphor.



Key

1 viewing window

a Moving the content by the “move” command from a user.

b Content to be viewed.

c New content presented through the viewing window.

Figure 5 — Movement of content

7.2 Consideration of the movement metaphor

The movement metaphor is used in various ICT systems and applications. The systems and the applications utilize the metaphor depending on their features. When a user manipulates a specific object, the “movement of content” metaphor is applied to the content. On the other hand, when the user navigates a whole object, the “movement of a window” metaphor is applied to the navigation of the whole content. In order to help the user to understand on which type of the movement metaphor he is utilizing, the system should indicate what is currently manipulated (using a focus or other audio/visual effects).

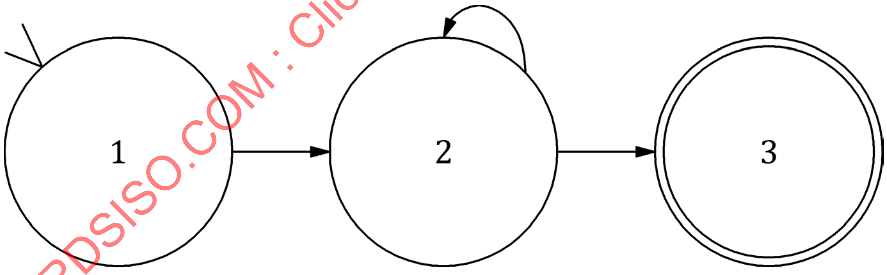
It follows that a gesture of moving a POI generates different results according to its movement metaphor. This should be considered when the gestures and their associated functions are discussed in this document.

8 Descriptions of the single-point gestures

8.1 General

Two types of gestures are defined in this document: the “directional” gestures and the “command” gestures. The directional gestures are formulated using directional movements of a POI. The movements denote directional information of the gestures such as “moving an object to a left direction”. The command gestures denote specific meanings of commands such as “initiation of an application”, “termination of an application” and so on.

For each gesture in this document, an initial state, an intermediate state and a final state shall be defined. A starting point of a gesture (where a user's POI activates the gesture and it is recognized by an ICT system or an application) denotes an initial state. An ending point of a gesture (where the gesture terminates and is recognized) represents a final state. There shall be a clear distinction between a starting point and an ending point of each single-point gesture for the clear recognition of gestures. An intermediate state is expressed by movements between a starting point and an ending point. Therefore, a gesture-based interface shall recognize and classify the user’s movements as an initial state, an intermediate state, and a final state of a gesture.



Key

1 initial state
2 intermediate state
3 final state

Figure 6 — State transition diagram of the gestures

The gestures in this document are expressed using the template which is defined in ISO/IEC 30113-1:2015, A.3.5. The state transition diagram in Figure 6 is referred by the descriptions of the gestures in this document.

8.2 Directional gestures

8.2.1 General

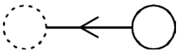
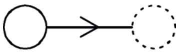
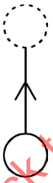
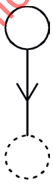
The most primitive single-point gestures are the directional gestures. As summarized in Table 1, gestures of denoting four directional movements of a POI (“left”, “right”, “up”, and “down”) are the simple directional gestures.

If a user performs a mediated gesture using a mouse, for example, he/she presses a right button of the mouse to start the gesture. Then, a POI should be moved to one of the four directions using the mouse. If the right button is released, the gesture is completed.

In case of a direct touch gesture, when a user touches a screen surface, a gesture recognition component recognizes the action as an initial state of a gesture. As the user moves his/her fingertip, one of the four directional movements can be recognized. If the user disconnects his/her fingertip from the screen, the gesture is completed.

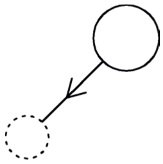
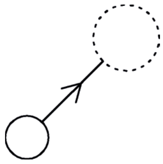
A non-contact gesture can also be used in performing the gestures for four-directional movements of a POI. The gestures can be executed on a virtual “x-y” plane.

Table 1 — Gestures of four-directional movements

ID	Name	Visual description	Explanation
G11-1	Left		Moving a POI horizontally along the left direction.
G11-2	Right		Moving a POI horizontally along the right direction.
G11-3	Up		Moving a POI vertically along the up direction.
G11-4	Down		Moving a POI vertically along the down direction.

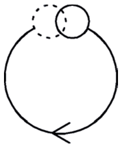
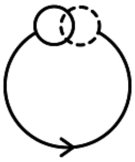
It is noted that a directional gesture can be performed in a 3D space. The “Forward” gesture and the “Backward” gesture can be used in moving a POI forward and backward along the “z” axis, respectively (see Table 2). In this case, a symbol denoting a POI is used in the visual description of the gesture in order to describe the “z”-directional movement in a clearer way.

Table 2 — Gestures of directional movements along the "z" axis

ID	Name	Visual description	Explanation
G11-5	Forward		Pushing a POI from the current position along the forward direction (i.e. moving the POI away from a user along the z axis). Its starting point (the solid circle) is drawn bigger than its ending point (the dotted circle), in order to visually express the "forward" moving direction.
G11-6	Backward		Pulling a POI from the current position along the forward direction (i.e. moving the POI toward a user along the z axis). Its ending point (the dotted circle) is drawn bigger than its starting point (the solid circle), in order to visually express the "backward" moving direction.

The circular movements presented in Table 3 are regarded as single-point gestures. The direction of the movements for drawing a circle can be either clockwise or counter-clockwise.

Table 3 — Gestures of circular movements

ID	Name	Visual description	Explanation
G11-7	Clockwise rotation		Drawing a circle in a 2D plane by moving a POI in a clockwise direction. The starting point and the ending point shall meet and make a circle. If so, the positions of the points on the circle do not matter.
G11-8	Counter-clockwise rotation		Drawing a circle in a 2D plane by moving a POI in a counter-clockwise direction. The starting point and the ending point shall meet and make a circle. If so, the positions of the points on the circle do not matter.

8.2.2 "Left" gesture

- **Unique (internal) identifier**: G11-1.
- **Text name of the gesture**: Left.
- **Text description of the gesture**: a gesture of moving a POI horizontally along a left direction.
- **Key or mouse alternative**: a stroke of a left arrow key should be regarded as the correspondence of the gesture.
- **Graphic representation of the gesture**: see Figure 7.

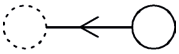


Figure 7 — Moving the POI to the left direction

- **Number of states involved in the gesture**: 3.
- **Initial state**:
  - order identifier of the state: 1;
  - starting position(s): inside of a space where a user shall perform the gesture to be recognized by an input device;

  - movement(s) or condition(s): the input device shall be in a mode where actions of the user can be recognized and the POI shall be recognized by the device.

- **Intermediate state**:
  - order identifier of the state: 2;
  - starting position(s): the position of the POI in the initial state;
  - movement(s) or condition(s): moving the POI horizontally along the left direction;
  - permitted variations: any movement within 30° of a horizontal axis along the left direction shall be regarded as the gesture. The 30° constraint is an example. The value can vary depending on an application. Its visual description should be presented as shown in Figure 8.

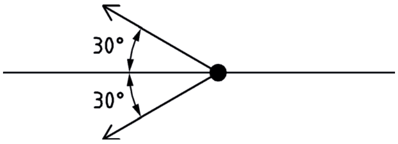


Figure 8 — Permitted variations for the Left gesture

- **Final state**:
  - order identifier of the state: 3;
  - starting position(s): the final position of the POI after the state 2 (the intermediate state);
  - movement(s) or condition(s): the user finishes the gesture.
- **Graphic representation of the states**: See Figure 6.
- **Specific instances of the gesture**: The following specific instances shall be considered. The Left gesture should be considered as various commands depending on its contexts or circumstances.
  - **Moving an object to a left direction**: a user wants to move a selected object to a left direction relative to a background. If an icon on a screen is selected, the Left gesture should make the user move the icon to the left direction.
  - **Scrolling to a left direction**: a user moves a viewing window to a left direction of content, if the window does not display the whole content. The "movement of a viewing window" metaphor is used in this case.
  - **Going back to an upper level of a menu**: during navigating a menu which has a multiple levels, a user wants to go back to the upper level of the menu hierarchy. The gesture should be utilized to return back to the upper level.
  - **Going to a previous application**: when a user checks a history of executing applications, he/she wants to move to a previously executed application. The gesture should be used to execute the function.
  - **Visiting to a previous page**: when a user visits a Web site, he/she wants to move to a previously visited Web page. The gesture should be used to execute the function.

  NOTE The Left gesture can be interpreted as a function for visiting a next page in some culture.

  - **Changing a device channel to a previous one**: when a user controls a device, such as smart TV, he/she wants to change a channel of the device to a previous channel. The gesture should be used to execute the function.

- **Turing a viewpoint to a left direction in a 3D environment**: if a user walks through a 3D virtual environment, for example, he/she wants to look around the environment by turning a viewpoint. The Left gesture should be used to turn the viewpoint to a left direction.

8.2.3 "Right" gesture

- **Unique (internal) identifier**: G11-2.
- **Text name of the gesture**: Right.
- **Text description of the gesture**: a gesture of moving a POI horizontally along a right direction.
- **Key or mouse alternative**: a stroke of a right arrow key should be regarded as the correspondence of the gesture.
- **Graphic representation of the gesture**: see Figure 9.

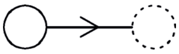


Figure 9 — Moving the POI to the right direction

- **Number of states involved in the gesture**: 3.
- **Initial state**:
  - order identifier of the state: 1;
  - starting position(s): inside of a space where a user shall perform the gesture to be recognized by an input device;
  - movement(s) or condition(s): the input device shall be in a mode where actions of the user can be recognized and the POI shall be recognized by the device.
- **Intermediate state:**
  - order identifier of the state: 2;
  - starting position(s): the position of the POI in the initial state;
  - movement(s) or condition(s): moving the POI horizontally along a right direction;
  - permitted variations: any movement within 30° of a horizontal axis along the right direction shall be regarded as the gesture. The 30° constraint is an example. The value can vary depending on an application. Its visual description should be presented as shown in Figure 10.

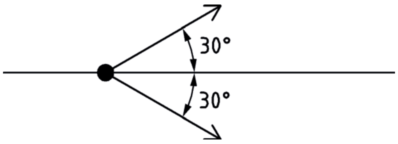


Figure 10 — Permitted variations of the Right gesture

- **Final state:**
  - order identifier of the state: 3;
  - starting position(s): the final position of the POI after the state 2 (the intermediate state);

- movement(s) or condition(s): the user finishes the gesture.

- **Graphic representation of the states:** see Figure 6.
- **Specific instances of the gesture**: The Right gesture shall be considered as various commands depending on its contexts or circumstances. The following specific instances should be considered.
  - **Moving an object to a right direction**: A user wants to move a selected object to a right direction relative to a background. If an icon on a screen is selected, the Right gesture should make the user move the icon to the right direction.
  - **Scrolling to a right direction**: A user moves a viewing window to a right direction of content, if the window does not display the whole content. The "movement of a viewing window" metaphor is used in this case.
  - **Going to a next application**: When a user checks a history of executing applications, he/she wants to move to a next application in the list. The gesture should be used to execute the function.
  - **Visiting to a next page**: When a user visits a Web site, he/she wants to move to a next webpage. The gesture should be used to execute the function.

  NOTE The Right gesture can be interpreted as a function for visiting a previous page in some culture.

  - **Changing a device channel to a next one**: When a user controls a device such as smart TV, he/she wants to change a channel of the device to a next channel. The gesture should be used to execute the function.
  - **Turning a viewpoint to a right direction in a 3D environment**: If a user walks through a 3D virtual environment, for example, he/she wants to look around the environment by turning a viewpoint. The Right gesture should be used to turn the viewpoint to a right direction.

8.2.4 “Up” gesture

- **Unique (internal) identifier**: G11-3.
- **Text name of the gesture**: Up.
- **Text description of the gesture**: a gesture of moving a POI vertically along an up direction.
- **Key or mouse alternative**: a stroke of up arrow key should be regarded as a correspondence of the gesture.
- **Graphic representation of the gesture**: see Figure 11.

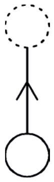


Figure 11 — Moving the POI to the up direction

- **Number of states involved in the gesture**: 3.
- **Initial state**:
  - order identifier of the state: 1;
  - starting position(s): inside of a space where a user shall perform the gesture to be recognized by an input device;

- movement(s) or condition(s): the input device shall be in a mode where actions of the user can be recognized and the POI shall be recognized by the device.

- **Intermediate state:**
  - order identifier of the state: 2;
  - starting position(s): the position of the POI in the initial state;
  - movement(s) or condition(s): moving the POI vertically along an up direction;
  - permitted variations: any movement within 30° of a vertical axis along an up direction shall be regarded as the gesture. The 30° constraint is an example. The value can vary depending on an application. Its visual description should be presented as shown in Figure 12.

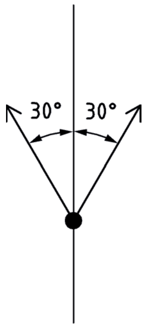


Figure 12 — Permitted variations of the Up gesture

- **Final state:**
  - Order identifier of the state: 3;
  - Starting position(s): the final position of the POI after the state 2 (the intermediate state);
  - Movement(s) or condition(s): the user finishes the gesture.
- **Graphic representation of the states:** see Figure 6.
- **Specific instances of the gesture**: The Up gesture shall be considered as various commands depending on its contexts or circumstances. The following specific instances should be considered.
  - **Moving an object to an up direction**: A user wants to move a selected object to an up direction relative to a background. If a corresponding icon is selected, the up gesture should make the user move the icon to the upper position.
  - **Scrolling to an up direction**: A user moves a viewing window to an up direction of content, if the window does not display the whole content. The "movement of a viewing window" metaphor is used in this case.
  - **Visiting to a previous viewport in a document**: When a user visits a webpage with a document, he/she wants to move a POI up to a previous viewport in the page. The gesture should be used to execute the function.
  - **Going back to an upper item of a menu**: During navigating a menu which has multiple menu items in a level, a user wants to move a POI up to an upper item of the menu level. The gesture should be utilized to go up to the upper menu item in the same level.
  - **Changing a device volume up**: When a user controls a device such as Smart TV, he/she wants to make a sound of the device louder. It means a volume of the device shall be higher. The gesture should be used to execute the function.

- **Turning a viewpoint to an up direction in a 3D environment**: If a user walks through a 3D virtual environment, for example, he/she wants to look around the environment by turning a viewpoint. In such a case, a POI should denote the viewpoint of the user. The "Up" gesture should be used to turn the viewpoint to an up direction.

8.2.5 "Down" gesture

- **Unique (internal) identifier**: G11-4.
- **Text name of the gesture**: Down.
- **Text description of the gesture**: a gesture of moving a POI vertically along a down direction.
- **Key or mouse alternative**: a stroke of a "down" arrow key should be regarded as a correspondence of the gesture.
- **Graphic representation of the gesture**: see Figure 13.

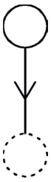


Figure 13 — Moving the POI to the down direction

- **Number of states involved in the gesture**: 3.
- **Initial state**:
  - order identifier of the state: 1;
  - starting position(s): inside of a space where a user shall perform the gesture to be recognized by an input device;
  - movement(s) or condition(s): the input device shall be in a mode where actions of the user can be recognized and the POI shall be recognized by the device.
- **Intermediate state**:
  - order identifier of the state: 2;
  - starting position(s): the position of the POI in the initial state;
  - movement(s) or condition(s): moving the POI vertically along a down direction;
  - permitted variations: any movement within 30° of a vertical axis along a down direction shall be regarded as the gesture. The 30° constraint is an example. The value can vary depending on an application. Its visual description should be presented as shown in Figure 14.

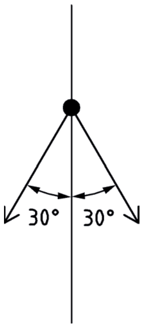


Figure 14 — Permitted variations of the Down gesture

- **Final state**:
  - order identifier of the state: 3;
  - starting position(s): the final position of the POI after the state 2 (the intermediate state);
  - movement(s) or condition(s): the user finishes the gesture.
- **Graphic representation of the states**: see Figure 6.
- **Specific instances of the gesture**: The "Down" gesture shall be considered as various commands depending on its contexts or circumstances. The following specific instances should be considered.
  - **Moving an object to a down direction**: A user wants to move a file to a down direction relative to a background. If a corresponding file icon is selected, the "Down" gesture should make the user move the file to the desired position.
  - **Scrolling to a down direction**: A user moves a viewing window to a down direction of content, if the window does not display the whole content. The "movement of a viewing window" metaphor is used in this case.
  - **Visiting to a next page in a document**: When a user visits a webpage with a document, he/she wants to move a POI to a next page in the document. The gesture should be used to execute the function.
  - **Going back to a lower item of a menu**: During navigating a menu which has multiple items in a level, a user wants to move a POI to a lower item of the menu. The gesture should be utilized to go up to the lower menu item in the same level.
  - **Changing a device volume down**: When a user controls a device such as smart TV, he/she wants to make a sound of the device smaller. It means a volume of the device shall become lower. The gesture should be used to execute the function.
  - **Turning a viewpoint to a down direction in a 3D environment**: If a user walks through a 3D virtual environment, for example, he/she wants to look around the environment by turning a viewpoint. In such a case, a POI should denote the viewpoint of the user. The "Down" gesture should be used to turn the viewpoint to a down direction.

8.2.6 "Forward" gesture

- **Unique (internal) identifier**: G11-5.
- **Text name of the gesture**: Forward.
- **Text description of the gesture**: a gesture of pushing a POI along the "z" axis, which is a direction of moving the POI farther from a user.

- **Key or mouse alternative**: a user with a wheel mouse should rotate a wheel of the mouse along a forward direction.
- **Graphic representation of the gesture**: see Figure 15.

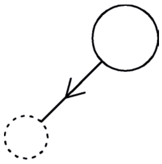
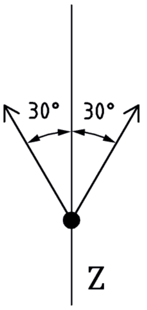


Figure 15 — Moving the POI to the forward direction

- **Number of states involved in the gesture**: 3.
- **Initial state**:
  - order identifier of the state: 1;
  - starting position(s): inside of a space where a user shall perform the gesture to be recognized by an input device;
  - movement(s) or condition(s): the input device shall be in a mode where actions of the user can be recognized and the POI shall be recognized by the device.
- **Intermediate state**:
  - order identifier of the state: 2;
  - starting position(s): the position of the POI in the initial state;
  - movement(s) or condition(s): to move the POI farther from a user along a “z” axis;
  - permitted variations: any push movement within 30° of a “z” axis farther from a user shall be regarded as the gesture. The 30° constraint is an example. The value can vary depending on an application. Its visual description should be presented as shown in Figure 16.



Key

Z z-axis

Figure 16 — Permitted variations of the Forward gesture

- **Final state**:
  - order identifier of the state: 3;
  - starting position(s): the final position of the POI after the state 2 (the intermediate state);

- movement(s) or condition(s): the user finishes the gesture.

- **Graphic representation of the states**: see Figure 6.
- **Specific instances of the gesture**: The "Forward" gesture shall be considered as various commands depending on its contexts or circumstances. The following specific instances should be considered.
  - **Moving an object farther from a user**: A user wants to move a virtual object farther from the user in a 3D virtual environment. If the object is selected, the "Forward" gesture should make the user move the object to a desired position.
  - **Moving an avatar forward**: A user wants to move his/her avatar forward a 3D virtual environment. If a "travel" or "walking" mode is selected during a 3D navigation of the virtual environment, the "Forward" gesture should make the user move the avatar forward to a desired position.
  - **Pushing a viewpoint farther from a user in a 3D virtual environment**: If a user walks through a 3D virtual environment, for example, he/she wants to look around the environment by turning a viewpoint. In such a case, a POI can denote the viewpoint of the user. The "Forward" gesture should be used to push the viewpoint to a farther position from the user.

8.2.7 "Backward" gesture

- **Unique (internal) identifier**: G11-6.
- **Text name of the gesture**: Backward.
- **Text description of the gesture**: a gesture of pulling a POI along the "z" axis, which is a direction of moving the POI closer to the user.
- **Key or mouse alternative**: a user with a wheel mouse should rotate the wheel of the mouse along a backward direction.
- **Graphic representation of the gesture**: see Figure 17.

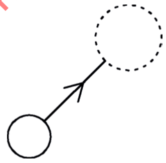
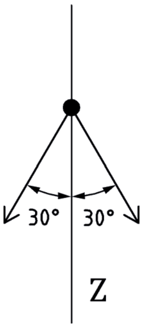


Figure 17 — Moving the POI to the backward direction

- **Number of states involved in the gesture**: 3.
- **Initial state**:
  - order identifier of the state: 1;
  - starting position(s): inside of a space where a user shall perform the gesture to be recognized by an input device;
  - movement(s) or condition(s): the input device shall be in a mode where actions of the user can be recognized and the POI shall be recognized by the device.
- **Intermediate state:**
  - order identifier of the state: 2;
  - starting position(s): the position of the POI in the initial state;
  - movement(s) or condition(s): moving the POI closer to the user along the "z" axis;

- permitted variations: any pulling movement within 30° of a "z" axis along a backward direction shall be regarded as the gesture. The 30° constraint is an example. The value can vary depending on an application. Its visual description should be presented as shown in Figure 18.



Key

Z	z-axis

Figure 18 — Permitted variations of the Backward gesture

- **Final state**:
  - order identifier of the state: 3;
  - starting position(s): the final position of the POI after the state 2 (the intermediate state);
  - movement(s) or condition(s): the user finishes the gesture.
- **Graphic representation of the states**: see Figure 6.
- **Specific instances of the gesture**: The "Backward" gesture shall be considered as various commands depending on its contexts or circumstances. The following specific instances should be considered.
  - **Moving an object closer to a user**: A user wants to move a virtual object closer to the user in a 3D virtual environment. If the object is selected, the "Backward" gesture should make the user move the object to a desired position.
  - **Moving an avatar backward**: A user wants to move his/her avatar backward in a 3D virtual environment. If a "travel" or "walking" mode is selected during a 3D navigation of the virtual environment, the "Backward" gesture should make the user move the avatar backward to a desired position.
  - **Pulling a viewpoint closer to a user in a 3D virtual environment**: If a user walks through a 3D virtual environment, for example, he/she wants to look around the virtual environment by turning his/her viewpoint. In such a case, a POI can denote the viewpoint of the user. The "Backward" gesture should be used to pull the viewpoint closer to the user.

8.2.8 "Clockwise rotation" gesture

- **Unique (internal) identifier**: G11-7.
- **Text name of the gesture**: Clockwise Rotation.
- **Text description of the gesture**: a gesture of drawing a circle with a POI along a clockwise direction. The positions of its starting point and its ending point can be anywhere on the circle as long as the two points meet and make the circle which is recognized by the system as the gesture.
- **Graphic representation of the gesture**: see Figure 19.

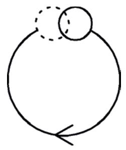


Figure 19 — Rotating the POI along a clockwise direction

- **Number of states involved in the gesture**: 3.
- **Initial state**:
  - order identifier of the state: 1;
  - starting position(s): inside of a space where a user shall perform the gesture to be recognized by an input device;
  - movement(s) or condition(s): the input device shall be in a mode where actions of the user can be recognized and the POI shall be recognized by the device.
- **Intermediate state**:
  - order identifier of the state: 2;
  - starting position(s): the position of the POI in the initial state;
  - movement(s) or condition(s): rotating the POI by drawing a circle along a clockwise direction on a 2D plane;
  - permitted variations: a circle which is drawn by a user shall be recognized as a closed curve in a 2D plane. Its starting point and ending point may not be exactly coincided at a same position.
- **Final state**:
  - order identifier of the state: 3;
  - starting position(s): the final position of the POI after the state 2 (the intermediate state);
  - movement(s) or condition(s): the user finishes the gesture.
- **Graphic representation of the states**: see Figure 6.
- **Specific instances of the gesture**: The "Clockwise rotation" gesture shall be considered as various commands depending on its contexts or circumstances. If anyone of the following operations is implemented, its specified gesture should be applied as follows.
  - **Rotating an object along a clockwise direction**: A user may want to rotate an object in a 2D or 3D environment along a clockwise direction relative to a background. If the object is selected, the "Clockwise rotation" gesture should make the user rotate the object along the clockwise direction.
  - **Rotating a viewpoint along a clockwise direction**: If a user navigates through a 2D or 3D environment, for example, he/she may want to look around the environment by rotating a viewpoint. In such a case, a POI should denote the viewpoint of the user. The "Clockwise rotation" gesture should be used to rotate the viewpoint along the clockwise direction.

8.2.9 "Counter-clockwise rotation" gesture

- **Unique (internal) identifier**: G11-8.
- **Text name of the gesture**: Counter-Clockwise Rotation.

- **Text description of the gesture**: a gesture of drawing a circle with a POI along a counter-clockwise direction. The positions of its starting point and its ending point can be anywhere on the circle as long as the two points meet and make the circle which is recognized by the system as the gesture.
- **Graphic representation of the gesture**: see Figure 20.

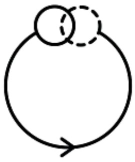


Figure 20 — Rotating the POI along a counter-clockwise direction

- **Number of states involved in the gesture**: 3.
- **Initial state**:
  - order identifier of the state: 1;
  - starting position(s): inside of a space where a user shall perform the gesture to be recognized by an input device;
  - movement(s) or condition(s): the input device shall be in a mode where actions of the user can be recognized and the POI shall be recognized by the device.
- **Intermediate state**:
  - order identifier of the state: 2;
  - starting position(s): the position of the POI in the initial state;
  - movement(s) or condition(s): rotating the POI by drawing a circle along a counter-clockwise direction on a 2D plane;
  - permitted variations: a circle which is drawn by a user shall be recognized as a closed curve in a 2D plane. Its starting point and ending point may not be exactly coincided at a same position.
- **Final state**:
  - order identifier of the state: 3;
  - starting position(s): the final position of the POI after the state 2 (the intermediate state);
  - movement(s) or condition(s): the user finishes the gesture.
- **Graphic representation of the states**: see Figure 6.
- **Specific instances of the gesture**: The “Counter-clockwise rotation” shall be considered as various commands depending on its contexts or circumstances. If anyone of the following operations is implemented, its specified gesture should be applied as follows.
  - **Rotating an object along a counter-clockwise direction**: A user may want to rotate an object in a 2D or 3D environment along a counter-clockwise direction relative to a background. If the object is selected, the “Counter-clockwise rotation” gesture should make the user rotate the object along the counter-clockwise direction.
  - **Rotating a viewpoint along a counter-clockwise direction**: If a user navigates through a 2D or 3D environment, for example, he/she may want to look around the environment by rotating a

viewpoint. In such a case, a POI should denote a viewpoint of the user. The “Counter-clockwise rotation” gesture should be used to rotate the viewpoint along the counter-clockwise direction.

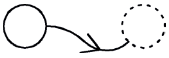
8.3 Command gestures

8.3.1 General

Gestures can invoke commands representing functions of activating applications, selecting an object, manipulating an object, and so on. The gestures are generally mapped to their corresponding commands which are intended to execute the functions.

As summarized in Table 4, a set of gestures are used in manipulating an object. A user should select an object using the “Tap” gesture (G11-9). A “drag and drop” action should be implemented by a gesture of tapping an object, displacing the object and releasing it at its destination. The gesture is defined as the “Displace” gesture (G11-10). If an object represents an application, a gesture of double-tapping the object (G11-11, the “Double-tap” gesture) should be interpreted as a function of activating the application. Such gestures are regarded as command gestures.

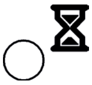
Table 4 — Gestures of manipulating an object

ID	Name	Visual description	Explanation
G11-9	Tap	○	Tapping an object with a POI.
G11-10	Displace		Moving a selected object to its destination. This gesture shall always be preceded by the “Tap” gesture which should be used to select the object. Its path from the starting point to the ending point can be arbitrary depending on their relative positions.
G11-11	Double-tap	◎	Double-tapping an icon or an object.

In Table 5, the “Tap and hold” gesture is described. The gesture involves maintaining the position of a POI for a certain amount of time.

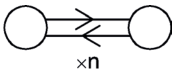
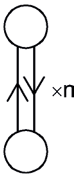
EXAMPLE One of possible functions for the “tap and hold” gesture (G11-12) is a command of “invoking a pop-up menu” which should be activated by clicking a right button of a mouse in Microsoft Windows. This gesture should also be performed using a direct touch gesture (as discussed in the 5.2), involving a “touch and holding” action with a certain amount of time.

Table 5 — Gesture of tapping and holding a POI

ID	Name	Visual description	Explanation
G11-12	Tap and hold		Tapping and holding a POI for a specific amount of time.

Another type of a gesture is a gesture with a waving action. Table 6 lists the gestures, each of which consists of the waving actions with different directions. The single-point gestures should be executed by moving a POI horizontally or vertically.

Table 6 — Gestures of waving

ID	Name	Visual description	Explanation
G11-13	Horizontal wave		Waving a POI from the left to the right and vice versa. Its starting point could be located at any position on the "horizontal waving" path.
G11-14	Vertical wave		Waving a POI from the up to the down and vice versa. Its starting point could be located at any position on the "vertical waving" path.

8.3.2 “Tap” gesture

- **Unique (internal) identifier**: G11-9.
- **Text name of the gesture**: Tap.
- **Text description of the gesture**: a gesture of selecting an object using a POI.
- **Key or mouse alternative**: a click of a mouse should be regarded as a correspondence of the gesture.
- **Graphic representation of the gesture**: see Figure 21.

Figure 21 — Tapping the POI

- **Number of states involved in the gesture**: 3.
- **Initial state**:
  - order identifier of the state: 1;
  - starting position(s): inside of a space where a user shall perform the gesture to be recognized by an input device;
  - movement(s) or condition(s): the input device shall be in a mode where actions of the user can be recognized and the POI shall be recognized by the device.
- **Intermediate state**:
  - order identifier of the state: 2;
  - starting position(s): the position of the POI in the initial state;
  - movement(s) or condition(s): tapping the POI without any movement;
  - permitted variations: any movement or a vibration of the POI within a specific small area, which shall be dependent on the recognition technique.
- **Final state**:
  - order identifier of the state: 3;
  - starting position(s): the final position of the POI after the state 2 (the intermediate state);

  - movement(s) or condition(s): the user finishes the gesture by a "releasing" action such as depressing a mouse button or disconnecting a fingertip from a screen or a surface.

- **Graphic representation of the states**: see Figure 6.

- **Specific instances of the gesture**: The "Tap" gesture shall be considered as various commands depending on its contexts or circumstances. The following specific instances should be considered.

  - **Selecting an icon:** user should utilize the "Tap" gesture to pick an icon which may represent an object or an application.

  - **Selecting an option:** user should utilize the "Tap" gesture to choose an option from a menu.

8.3.3 "Displace" gesture

- **Unique (internal) identifier**: G11-10.

- **Text name of the gesture**: Displace.

- **Text description of the gesture**: a gesture of displacing a selected object using a POI to a destination. The path of the gesture for displacing the POI should be arbitrary depending on its ending point.

- **Key or mouse alternative**: a "drag and drop" operation of a selected object with a mouse should be regarded as a correspondence of the gesture with the selected object.

- **Graphic representation of the gesture**: see Figure 22.

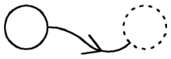


Figure 22 — Displacing the POI to a specific position

- **Number of states involved in the gesture**: 3.

- **Initial state**:

  - order identifier of the state: 1;

  - starting position(s): inside of a space where a user shall perform the gesture to be recognized by an input device;

  - movement(s) or condition(s): the input device shall be in a mode where actions of the user can be recognized and the POI shall be recognized by the device.

- **Intermediate state**:

  - order identifier of the state: 2;

  - starting position(s): the position of the POI in the initial state;

  - movement(s) or condition(s): moving the POI (which points the selected object) to a designated position;

  - permitted variations: any movement or a vibration of the POI within a specific small area, which shall be dependent on the recognition technique.

- **Final state**:

  - order identifier of the state: 3;

  - starting position(s): the final position of the POI after the state 2 (the intermediate state);

  - movement(s) or condition(s): the user finishes the gesture.
- **Graphic representation of the states**: see Figure 6.
- **Specific instances of the gesture**: The "Displace" gesture (which is applied to a selected object) shall be considered as a "drag and drop" function which should be interpreted as various commands depending on its contexts or circumstances. The following specific instances should be considered.
  - **Moving an object:** user should utilize the "Displace" gesture to move an object which may represent an object or an application. In a file system, a file icon shall be moved from a folder to another folder by the "Displace" gesture.
  - **Deleting an object:** user should delete an object by moving the object icon to a trash can folder or a garbage bin icon.

8.3.4 "Double-tap" gesture

- **Unique (internal) identifier**: G11-11.
- **Text name of the gesture**: Double-tap.
- **Text description of the gesture**: a gesture of double-tapping an object using a POI.
- **Key or mouse alternative**: double-click of a mouse should be regarded as a correspondence of the gesture.
- **Graphic representation of the gesture**: see Figure 23.

Figure 23 — Double-tapping the POI

- **Number of states involved in the gesture**: 3.
- **Initial state**:
  - order identifier of the state: 1;
  - starting position(s): inside of a space where a user shall perform the gesture to be recognized by an input device;
  - movement(s) or condition(s): the input device shall be in a mode where actions of the user can be recognized and the POI shall be recognized by the device.
- **Intermediate state:**
  - order identifier of the state: 2;
  - starting position(s): the position of the POI in the initial state;
  - movement(s) or condition(s): tapping the POI without any movement;
  - permitted variations: any movement or a vibration of the POI within a specific small area, which shall be dependent on the recognition technique.
- **Final state:**
  - order identifier of the state: 3;
  - starting position(s): the final position of the POI after the state 2 (the intermediate state);

  - movement(s) or condition(s): the user finishes the gesture.
- **Graphic representation of the states:** see Figure 6.
- **Specific instances of the gesture**: The “Double-tap” gesture shall be considered as various commands depending on its contexts or circumstances. The following specific instances should be considered:
  - **Opening an object:** A user should utilize the “Double-tap” gesture to pick and open an object such as a file.
  - **Executing an application:** A user should utilize the “Double-tap” gesture to pick and executing an application which is represented by an icon.

8.3.5 “Tap and hold” gesture

- **Unique (internal) identifier**: G11-12.
- **Text name of the gesture**: Tap and Hold.
- **Text description of the gesture**: a gesture of tap and holding a POI with a specific amount of time.
- **Key or mouse alternative**: clicking and pressing a button of a mouse should be regarded as a correspondence of the gesture.
- **Graphic representation of the gesture**: see Figure 24.

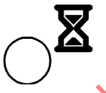


Figure 24 — Tapping and holding

- **Number of states involved in the gesture**: 3.
- **Initial state**:
  - order identifier of the state: 1;
  - starting position(s): inside of a space where a user shall perform the gesture to be recognized by an input device;
  - movement(s) or condition(s): the input device shall be in a mode where actions of the user can be recognized and the POI shall be recognized by the device.
- **Intermediate state:**
  - order identifier of the state: 2;
  - starting position(s): the position of the POI in the initial state;
  - movement(s) or condition(s): tap and holding the POI without any movement for a specific amount of time;
  - permitted variations: any movement or a vibration of the POI within a specific small area, which shall be dependent on the recognition technique.
- **Final state:**
  - order identifier of the state: 3;
  - starting position(s): the final position of the POI after the state 2 (the intermediate state);