

libsystemgesture Reference

© 2015 Sony Computer Entertainment Inc.
All Rights Reserved.
SCE Confidential

Table of Contents

Constant Definitions	3
Gesture Types	4
Touch Event States	5
Error Codes	6
Datatypes	7
SceSystemGestureVector2	8
SceSystemGestureRectangle	9
SceSystemGesturePrimitiveTouchEvent	10
SceSystemGesturePrimitiveTouchRecognizerParameter	11
SceSystemGestureTouchRecognizer	12
SceSystemGestureTouchRecognizerInformation	13
SceSystemGestureTouchRecognizerParameter	14
SceSystemGestureTapRecognizerParameter	15
SceSystemGestureDragRecognizerParameter	16
SceSystemGestureTapAndHoldRecognizerParameter	17
SceSystemGesturePinchOutInRecognizerParameter	18
SceSystemGestureRotationRecognizerParameter	19
SceSystemGestureTouchEvent	20
SceSystemGestureTapEventProperty	22
SceSystemGestureDragEventProperty	23
SceSystemGestureTapAndHoldEventProperty	24
SceSystemGesturePinchOutInEventProperty	25
SceSystemGestureRotationEventProperty	26
Functions	27
sceSystemGestureInitializePrimitiveTouchRecognizer	28
sceSystemGestureFinalizePrimitiveTouchRecognizer	29
sceSystemGestureResetPrimitiveTouchRecognizer	30
sceSystemGestureUpdatePrimitiveTouchRecognizer	31
sceSystemGestureGetPrimitiveTouchEvents	32
sceSystemGestureGetPrimitiveTouchEventsCount	33
sceSystemGestureGetPrimitiveTouchEventByIndex	34
sceSystemGestureGetPrimitiveTouchEventByPrimitiveID	35
sceSystemGestureCreateTouchRecognizer	37
sceSystemGestureResetTouchRecognizer	39
sceSystemGestureGetTouchRecognizerInformation	40
sceSystemGestureUpdateTouchRecognizer	41
sceSystemGestureUpdateTouchRecognizerRectangle	42
sceSystemGestureGetTouchEvents	43
sceSystemGestureGetTouchEventsCount	44
sceSystemGestureGetTouchEventByIndex	45
sceSystemGestureGetTouchEventByEventID	46

Constant Definitions

Gesture Types

Definitions of types of gestures supported by library

Definition

```
#include <systemgesture.h>
typedef enum SceSystemGestureType
{
    SCE_SYSTEM_GESTURE_TYPE_TAP                = 0x00000001,
    SCE_SYSTEM_GESTURE_TYPE_DRAG                = 0x00000002,
    SCE_SYSTEM_GESTURE_TYPE_TAP_AND_HOLD       = 0x00000004,
    SCE_SYSTEM_GESTURE_TYPE_PINCH_OUT_IN       = 0x00000008,
    SCE_SYSTEM_GESTURE_TYPE_ROTATION            = 0x00000010
} SceSystemGestureType;
```

Description

These constants are the definition values of the types of gestures supported by libsystemgesture. They are used when generating a touch gesture recognizer using `sceSystemGestureCreateTouchRecognizer()`.

Currently, five types of gestures are supported: Tap, Drag, Tap and Hold, Pinch Out/In, and Rotation. For details, refer to the description of `sceSystemGestureCreateTouchRecognizer()`.

See Also

`sceSystemGestureCreateTouchRecognizer()`

Touch Event States

Definitions of states in which primitive touch events and various touch events such as Tap can take place

Definition

```
#include <systemgesture.h>
typedef enum SceSystemGestureTouchState
{
    SCE_SYSTEM_GESTURE_TOUCH_STATE_INACTIVE = 0x00000000,
    SCE_SYSTEM_GESTURE_TOUCH_STATE_BEGIN   = 0x00000001,
    SCE_SYSTEM_GESTURE_TOUCH_STATE_ACTIVE   = 0x00000002,
    SCE_SYSTEM_GESTURE_TOUCH_STATE_END      = 0x00000003,
    SCE_SYSTEM_GESTURE_TOUCH_STATE_CANCELLED = 0x00000004
} SceSystemGestureTouchState;
```

Description

These constants are the definition values of the states in which primitive touch events and touch events such as Tap, Drag, and Tap and Hold can take place.

For details on primitive events, refer to `SceSystemGesturePrimitiveTouchEvent`, and for details on touch events such as Tap, refer to `SceSystemGestureTouchEvent`.

See Also

`SceSystemGesturePrimitiveTouchEvent`, `SceSystemGestureTouchEvent`

SCE CONFIDENTIAL

Error Codes

List of error codes

Definition

Value	Hexadecimal	Description
SCE_SYSTEM_GESTURE_ERROR_INVALID_ARGUMENT	0x80890001	An invalid argument was specified
SCE_SYSTEM_GESTURE_ERROR_NOT_SUPPORTED_GESTURE	0x80890002	An unsupported gesture was specified
SCE_SYSTEM_GESTURE_ERROR_NOT_INITIALIZED	0x80890003	The gesture recognizer was not initialized
SCE_SYSTEM_GESTURE_ERROR_INDEX_OUT_OF_ARRAY	0x80890004	The specified <i>index</i> exceeds the array range
SCE_SYSTEM_GESTURE_ERROR_EVENT_DATA_NOT_FOUND	0x80890005	No event data could be found

Datatypes

000004892117

SCE CONFIDENTIAL

SceSystemGestureVector2

Structure to which the touch position and travel amount are saved

Definition

```
#include <systemgesture.h>
typedef struct SceSystemGestureVector2
{
    SceInt16 x;
    SceInt16 y;
} SceSystemGestureVector2;
```

Members

x x coordinate component
y y coordinate component

Description

This is a structure to which the touch position and travel amount are saved. The touch panel coordinate system values are used for the *x* and *y* components.

See Also

SceSystemGesturePrimitiveTouchEvent, SceSystemGestureTouchEvent

SceSystemGestureRectangle

Structure to which the target rectangle of the touch gesture recognizer is stored

Definition

```
#include <systemgesture.h>
typedef struct SceSystemGestureRectangle
{
    SceInt16 x;
    SceInt16 y;
    SceInt16 width;
    SceInt16 height;
} SceSystemGestureRectangle;
```

Members

<i>x</i>	x coordinate of top-left point of rectangle
<i>y</i>	y coordinate of top-left point of rectangle
<i>width</i>	Width of rectangle
<i>height</i>	Height of rectangle

Description

This is a structure to which the target rectangle of the touch gesture recognizer is stored. The touch panel coordinate system values are used for the position and size.

See Also

`sceSystemGestureCreateTouchRecognizer()`,
`sceSystemGestureUpdateTouchRecognizerRectangle()`

SCE CONFIDENTIAL

SceSystemGesturePrimitiveTouchEvent

Structure for primitive touch events

Definition

```
#include <systemgesture.h>
typedef struct SceSystemGesturePrimitiveTouchEvent
{
    SceSystemGestureTouchState eventState;
    SceUInt16 primitiveID;

    SceSystemGestureVector2 pressedPosition;
    SceInt16 pressedForce;

    SceSystemGestureVector2 currentPosition;
    SceInt16 currentForce;

    SceSystemGestureVector2 deltaVector;
    SceInt16 deltaForce;

    SceUInt64 deltaTime;

    SceUInt64 elapsedTime;

    SceUInt8 reserved[56];
} SceSystemGesturePrimitiveTouchEvent;
```

Members

<i>eventState</i>	State of primitive touch event
<i>primitiveID</i>	ID of primitive touch event.
	The touch panel port number is stored in the top 8 bits
<i>pressedPosition</i>	Pressed position
<i>pressedForce</i>	Touch force used to press touch panel
<i>currentPosition</i>	Current touch position
<i>currentForce</i>	Force of current touch
<i>deltaVector</i>	Amount of travel from last data update
<i>deltaForce</i>	Change in touch force from last data update
<i>deltaTime</i>	Time elapsed since last data update (microsec)
<i>elapsedTime</i>	Time elapsed since press (microsecond)
<i>reserved</i>	Reserved area

Description

This is a structure for primitive touch events. There are four primitive touch events, namely pressing the touch panel with a finger (Begin), maintaining contact with the touch panel with a finger (Active), releasing the touch panel by lifting one's finger off the touch panel (End), and not touching the touch panel with a finger (Inactive).

See Also

```
sceSystemGestureGetPrimitiveTouchEvents(),
sceSystemGestureGetPrimitiveTouchEventByIndex(),
sceSystemGestureGetPrimitiveTouchEventByPrimitiveID()
```

SceSystemGesturePrimitiveTouchRecognizerParameter

Parameter for initializing primitive touch gesture recognizer

Definition

```
#include <systemgesture.h>
typedef struct SceSystemGesturePrimitiveTouchRecognizerParameter {
    SceUInt8 reserved[ 64 ];
} SceSystemGesturePrimitiveTouchRecognizerParameter;
```

Members

reserved Reserved area

Description

This is the parameter for `sceSystemGestureInitializePrimitiveTouchRecognizer()`.
It is provided for future expansion.

SCE CONFIDENTIAL

SceSystemGestureTouchRecognizer

Structure to which the work area of the touch gesture recognizer is stored

Definition

```
#include <systemgesture.h>
typedef struct SceSystemGestureTouchRecognizer
{
    SceUInt64 reserved[ 307 ];
} SceSystemGestureTouchRecognizer;
```

Members

reserved Reserved area

Description

This is a structure to which the work area of the touch gesture recognizer is stored. It is initialized using `sceSystemGestureCreateTouchRecognizer()`.

See Also

`sceSystemGestureCreateTouchRecognizer()`

SCE CONFIDENTIAL

SceSystemGestureTouchRecognizerInformation

Structure for getting the information of the touch gesture recognizer

Definition

```
#include <systemgesture.h>
typedef struct SceSystemGestureTouchRecognizerInformation
{
    SceSystemGestureType gestureType;
    SceUInt32 touchPanelPortID;
    SceSystemGestureRectangle rectangle;
    SceUInt64 updateTime;

    SceUInt8 reserved[ 256 ];
} SceSystemGestureTouchRecognizerInformation;
```

Members

<i>gestureType</i>	Type of gesture
<i>touchPanelPortID</i>	Touch panel port number for recognition
<i>rectangle</i>	Rectangle area for recognition
<i>updateTime</i>	Time of last update
<i>reserved</i>	Reserved area

Description

This is a structure for getting the information of the touch gesture recognizer. Acquisition is done with `sceSystemGestureGetTouchRecognizerInformation()`.

See Also

`sceSystemGestureCreateTouchRecognizer()`

SceSystemGestureTouchRecognizerParameter

Union of parameters for initialization of gesture recognizers

Definition

```
#include <systemgesture.h>
typedef union SceSystemGestureTouchRecognizerParameter
{
    SceUInt8 parameterBuf[ 64 ];

    SceSystemGestureTapRecognizerParameter tap;
    SceSystemGestureDragRecognizerParameter drag;
    SceSystemGestureTapAndHoldRecognizerParameter tapAndHold;
    SceSystemGesturePinchOutInRecognizerParameter pinchOutIn;
    SceSystemGestureRotationRecognizerParameter rotation;
} SceSystemGestureTouchRecognizerParameter;
```

Members

<i>parameterBuf</i>	Parameter buffer area
<i>tap</i>	Tap gesture recognizer parameter
<i>drag</i>	Drag gesture recognizer parameter
<i>tapAndHold</i>	Tap and Hold gesture recognizer parameter
<i>pinchOutIn</i>	Pinch Out/In gesture recognizer parameter
<i>rotation</i>	Rotation gesture recognizer parameter

Description

This is the union of the setting parameters for the initialization of gesture recognizers. It is used when creating gesture recognizers with `sceSystemGestureCreateTouchRecognizer()`. For the parameters that can be set for the various gesture recognizers, refer to the sections describing the structures of gesture recognizer parameters.

See Also

`sceSystemGestureCreateTouchRecognizer()`

SceSystemGestureTapRecognizerParameter

Structure to which the Tap gesture recognizer initialization parameter is stored

Definition

```
#include <systemgesture.h>
typedef struct SceSystemGestureTapRecognizerParameter
{
    SceUInt8 maxTapCount;

    SceUInt8 reserved[ 63 ];
} SceSystemGestureTapRecognizerParameter;
```

Members

<i>maxTapCount</i>	Maximum continuous count when Tap gestures are being continuously performed
<i>reserved</i>	Reserved area

Description

This is a structure to which the Tap gesture recognizer initialization parameter is stored. It is passed as an argument when `sceSystemGestureCreateTouchRecognizer()` is used. Set a value of 1 or greater for *maxTapCount*. If 0 is specified, it is considered that 1 was specified and the Tap gesture recognizer is initialized.

When 1 is set for *maxTapCount*, the system judges that a Tap gesture has occurred in an instant when the finger breaks off contact with the touch panel (the touch panel is released). When a value of 2 or greater is set for *maxTapCount*, for Tap gestures up to *maxTapCount* -1, a Tap gesture is judged to have occurred following the lapse of a preset time from the time when the touch panel is released. For example, when *maxTapCount* is set to 3 and the touch panel is tapped once or twice in succession, a Tap gesture event occurs following the lapse of the preset time from when the touch screen was released. If the touch panel is tapped three times in succession, a Tap gesture event occurs the instant the touch panel is released.

If the gesture recognizer is initialized by calling `sceSystemGestureCreateTouchRecognizer()` with specifying NULL to *touchRecognizerParameter* without using this structure, it is considered that 1 (time) was specified as the value of *maxTapCount* and the Tap gesture recognizer is initialized.

See Also

`SceSystemGestureTouchRecognizerParameter`,
`sceSystemGestureCreateTouchRecognizer()`

SCE CONFIDENTIAL

SceSystemGestureDragRecognizerParameter

Structure to which the Drag gesture recognizer initialization parameter is stored

Definition

```
#include <systemgesture.h>
typedef struct SceSystemGestureDragRecognizerParameter
{
    SceUInt8 reserved[ 64 ];

} SceSystemGestureDragRecognizerParameter;
```

Members

reserved Reserved area

Description

This is a structure to which the Drag gesture recognizer initialization parameter is stored. Currently no parameters can be set.

See Also

SceSystemGestureTouchRecognizerParameter,
sceSystemGestureCreateTouchRecognizer()

SCE CONFIDENTIAL

SceSystemGestureTapAndHoldRecognizerParameter

Structure to which the Tap and Hold gesture recognizer initialization parameter is stored

Definition

```
#include <systemgesture.h>
typedef struct SceSystemGestureTapAndHoldRecognizerParameter
{
    SceUInt64 timeToInvokeEvent;

    SceUInt8 reserved[ 56 ];
} SceSystemGestureTapAndHoldRecognizerParameter;
```

Members

<i>timeToInvokeEvent</i>	Time until a Tap and Hold event occurs after the touch panel is pressed (millisecond)
<i>reserved</i>	Reserved area

Description

This is a structure to which the Tap and Hold gesture recognizer initialization parameter is stored. Whether a Tap and Hold gesture occurred is judged when the time specified to *timeToInvokeEvent* has lapsed after the finger has pressed the touch panel.

If the gesture recognizer is initialized by calling `sceSystemGestureCreateTouchRecognizer()` with specifying NULL to *touchRecognizerParameter* without using this structure, it is considered that 1000 (milliseconds) was specified as the value of *timeToInvokeEvent* and the Tap and Hold gesture recognizer is initialized.

See Also

`SceSystemGestureTouchRecognizerParameter`,
`sceSystemGestureCreateTouchRecognizer()`

SCE CONFIDENTIAL

SceSystemGesturePinchOutInRecognizerParameter

Structure to which the Pinch Out/In gesture recognizer initialization parameter is stored

Definition

```
#include <systemgesture.h>
typedef struct SceSystemGesturePinchOutInRecognizerParameter
{
    SceUInt8 reserved[ 64 ];

} SceSystemGesturePinchOutInRecognizerParameter;
```

Members

reserved Reserved area

Description

This is a structure to which the Pinch Out/In gesture recognizer initialization parameter is stored.
Currently no parameters can be set.

See Also

SceSystemGestureTouchRecognizerParameter,
sceSystemGestureCreateTouchRecognizer()

SCE CONFIDENTIAL

SceSystemGestureRotationRecognizerParameter

Structure to which the Rotation gesture recognizer initialization parameter is stored

Definition

```
#include <systemgesture.h>
typedef struct SceSystemGestureRotationRecognizerParameter
{
    SceUInt8 reserved[ 64 ];

} SceSystemGestureRotationRecognizerParameter;
```

Members

reserved Reserved area

Description

This is a structure to which the Rotation gesture recognizer initialization parameter is stored. Currently no parameters can be set.

See Also

SceSystemGestureTouchRecognizerParameter,
sceSystemGestureCreateTouchRecognizer()

SceSystemGestureTouchEvent

Structure for touch gesture events

Definition

```
#include <systemgesture.h>
typedef struct SceSystemGestureTouchEvent
{
    SceUInt32 eventID;
    SceSystemGestureTouchState eventState;
    SceSystemGestureType gestureType;
    SceUInt8 padding[4];
    SceUInt64 updateTime;

    union {
        SceUInt8 propertyBuf[ 64 ];
        SceSystemGestureTapEventProperty tap;
        SceSystemGestureDragEventProperty drag;
        SceSystemGestureTapAndHoldEventProperty tapAndHold;
        SceSystemGesturePinchOutInEventProperty pinchOutIn;
        SceSystemGestureRotationEventProperty rotation;
    } property;

    SceUInt8 reserved[56];
} SceSystemGestureTouchEvent;
```

Members

<i>eventID</i>	Event ID of touch event (value of 1 or greater). The touch panel port number is stored in the top 8 bits
<i>eventState</i>	Touch event state
<i>gestureType</i>	Type of gesture
<i>padding</i>	Reserved area
<i>updateTime</i>	Time of last update
<i>property</i>	Union of gesture event properties
<i>reserved</i>	Reserved area

The *property* members are as follows.

<i>propertyBuf</i>	Buffer for storing properties
<i>tap</i>	Tap gesture event properties
<i>drag</i>	Drag gesture event properties
<i>tapAndHold</i>	Tap and Hold gesture event properties
<i>pinchOutIn</i>	Pinch Out/In gesture event properties
<i>rotation</i>	Rotation gesture event properties

Description

This is a structure for touch gesture events such as Tap and Drag.

Acquisition is done with `sceSystemGestureGetTouchEvents()`, etc., following recognition by calling `sceSystemGestureUpdateTouchRecognizer()` for a gesture recognizer.

For the properties of the various touch gesture events, refer to the descriptions of the properties of the various touch gesture events in the following sections.

The *eventState* transitions differ according to the type of gesture, as follows.

<i>gestureType</i>	<i>Possible eventState</i>
SCE_SYSTEM_GESTURE_TYPE_TAP	Active only
SCE_SYSTEM_GESTURE_TYPE_DRAG	The state transitions are Begin, Active, End, in this order. The Begin state is entered when the finger pressed the touch panel. Then, the Active state is maintained, and when the finger breaks contact with the touch panel or it travels outside the recognition target rectangle, the End state is entered. When Common Dialog, etc. are overlapped during the operation, the Cancelled state is entered.
SCE_SYSTEM_GESTURE_TYPE_TAP_AND_HOLD	The state transitions are Begin, Active, End, in this order. The Begin state is entered when the finger, once pressed on the touch panel, does not travel by more than the preset threshold value and a time exceeding set at initialization lapses. Then the Active state is maintained, and when the finger breaks contact with the touch panel or it travels beyond the preset threshold value, the End state is entered. When Common Dialog, etc. are overlapped during the operation, the Cancelled state is entered.
SCE_SYSTEM_GESTURE_TYPE_PINCH_OUT_IN	The state transitions are Begin, Active, End, in this order. The Begin state is entered when two fingers touch the recognition target rectangle. Then, the Active state is maintained, and when one of the two fingers breaks contact with the touch panel or travels outside the recognition target rectangle, the End state is entered. When Common Dialog, etc. are overlapped during the operation, the Cancelled state is entered.
SCE_SYSTEM_GESTURE_TYPE_ROTATION	

See Also

```
sceSystemGestureGetTouchEvents(), sceSystemGestureGetTouchEventsCount(),
sceSystemGestureGetTouchEventByIndex(),
sceSystemGestureGetTouchEventByEventID()
```

SceSystemGestureTapEventProperty

Tap gesture event properties

Definition

```
#include <systemgesture.h>
typedef struct SceSystemGestureTapEventProperty
{
    SceUInt16 primitiveID;
    SceSystemGestureVector2 position;
    SceUInt8 tappedCount;

    SceUInt8 reserved[ 57 ];
} SceSystemGestureTapEventProperty;
```

Members

<i>primitiveID</i>	ID of primitive touch event related to this Tap gesture event
<i>position</i>	Tapped position
<i>tappedCount</i>	Number of continuous taps
<i>reserved</i>	Reserved area

Description

This is a data structure to which the Tap gesture event properties are stored.

See Also

SceSystemGestureTouchEvent

SCE CONFIDENTIAL

SceSystemGestureDragEventProperty

Drag gesture event properties

Definition

```
#include <systemgesture.h>
typedef struct SceSystemGestureDragEventProperty
{
    SceUInt16 primitiveID;
    SceSystemGestureVector2 deltaVector;
    SceSystemGestureVector2 currentPosition;
    SceSystemGestureVector2 pressedPosition;

    SceUInt8 reserved[ 50 ];
} SceSystemGestureDragEventProperty;
```

Members

<i>primitiveID</i>	ID of primitive touch event related to this Drag gesture event
<i>deltaVector</i>	Travel amount from last update
<i>currentPosition</i>	Current touch position
<i>pressedPosition</i>	Pressed position
<i>reserved</i>	Reserved area

Description

This is a data structure to which the Drag gesture event properties are stored.

See Also

SceSystemGestureTouchEvent

SCE CONFIDENTIAL

SceSystemGestureTapAndHoldEventProperty

Tap and Hold gesture event properties

Definition

```
#include <systemgesture.h>
typedef struct SceSystemGestureTapAndHoldEventProperty
{
    SceUInt16 primitiveID;
    SceSystemGestureVector2 pressedPosition;

    SceUInt8 reserved[ 58 ];
} SceSystemGestureTapAndHoldEventProperty;
```

Members

<i>primitiveID</i>	ID of primitive touch event related to this Tap and Hold gesture event
<i>pressedPosition</i>	Pressed position
<i>reserved</i>	Reserved area

Description

This is a data structure to which the Tap and Hold gesture event properties are stored.

See Also

SceSystemGestureTouchEvent

SCE CONFIDENTIAL

SceSystemGesturePinchOutInEventProperty

Pinch Out/In gesture event properties

Definition

```
#include <systemgesture.h>
typedef struct SceSystemGesturePinchOutInEventProperty
{
    float scale;

    struct {
        SceUInt16 primitiveID;
        SceSystemGestureVector2 currentPosition;
        SceSystemGestureVector2 deltaVector;
        SceSystemGestureVector2 pairedPosition;
    } primitive[2];

    SceUInt8 reserved[ 32 ];
} SceSystemGesturePinchOutInEventProperty;
```

Members

scale Relative distance change of two fingers from time when the second of the two fingers constituting this Pinch Out/In gesture event was pressed

primitive Structure to which the data of the two fingers constituting this Pinch Out/In gesture event are stored

reserved Reserved area

The *primitive* members are as follows.

primitiveID ID of primitive touch event related to this Pinch Out/In gesture event

currentPosition Current touch positions of fingers constituting this Pinch Out/In gesture event

deltaVector Amount of travel from last update of the fingers constituting this Pinch Out/In gesture event

pairedPosition Touch position when the second of the two fingers constituting this Pinch Out/In gesture event was pressed

Description

This is the data structure to which the Pinch Out/In gesture event properties are stored.

See Also

SceSystemGestureTouchEvent

SceSystemGestureRotationEventProperty

Rotation gesture event properties

Definition

```
#include <systemgesture.h>
typedef struct SceSystemGestureRotationEventProperty
{
    float angle;

    struct {
        SceUInt16 primitiveID;
        SceSystemGestureVector2 currentPosition;
        SceSystemGestureVector2 deltaVector;
        SceSystemGestureVector2 pairedPosition;
    } primitive[2];

    SceUInt8 reserved[ 32 ];
} SceSystemGestureRotationEventProperty;
```

Members

angle Rotation angle (rad) of two fingers from time when the second of the two fingers constituting this Rotation gesture event was pressed
A positive value means a clockwise rotation.

primitive Structure to which the data of the two fingers constituting this Rotation gesture event are stored

reserved Reserved area

The *primitive* members are as follows.

primitiveID ID of primitive touch event related to this Rotation gesture event

currentPosition Current touch positions of fingers constituting this Rotation gesture event

deltaVector Amount of travel from last update of the fingers constituting this Rotation gesture event

pairedPosition Touch position when the second of the two fingers constituting this Rotation gesture event was pressed

Description

This is the data structure to which the Rotation gesture event properties are stored.

See Also

SceSystemGestureTouchEvent

Functions

000004892117

SCE CONFIDENTIAL

sceSystemGestureInitializePrimitiveTouchRecognizer

Initialize primitive touch gesture recognizer

Definition

```
#include <systemgesture.h>
int sceSystemGestureInitializePrimitiveTouchRecognizer (
    SceSystemGesturePrimitiveTouchRecognizerParameter* parameter
);
```

Calling Conditions

Multithread safe.

Arguments

parameter Parameter for initialization of primitive touch gesture recognizer
Not used currently. Specify NULL.

Return Values

Returns SCE_OK.

Description

This function initializes the primitive touch gesture recognizer. Call this function before starting to use libsystemgesture, such as when starting an application.

SCE CONFIDENTIAL

sceSystemGestureFinalizePrimitiveTouchRecognizer

End primitive touch gesture recognizer

Definition

```
#include <systemgesture.h>
int sceSystemGestureFinalizePrimitiveTouchRecognizer();
```

Calling Conditions

Multithread safe.

Arguments

None

Return Values

Returns SCE_OK.

Description

This function executes the end processing for the primitive touch gesture recognizer. It clears the area used inside the library.

SCE CONFIDENTIAL

sceSystemGestureResetPrimitiveTouchRecognizer

Reset primitive touch gesture recognizer

Definition

```
#include <systemgesture.h>
int sceSystemGestureResetPrimitiveTouchRecognizer();
```

Calling Conditions

Multithread safe.

Arguments

None

Return Values

Value	Description
SCE_OK	Normal completion
SCE_SYSTEM_GESTURE_ERROR_NOT_INITIALIZED	The primitive touch gesture recognizer was not initialized

Description

This function returns the primitive touch gesture recognizer to its initial state. Call this function to analyze newly a touch event, for example when the scene in a game application has changed.

See Also

sceSystemGestureInitializePrimitiveTouchRecognizer()

SCE CONFIDENTIAL

sceSystemGestureUpdatePrimitiveTouchRecognizer

Update primitive touch gesture recognizer

Definition

```
#include <systemgesture.h>
int sceSystemGestureUpdatePrimitiveTouchRecognizer (
    const SceTouchData* pFrontData,
    const SceTouchData* pBackData
);
```

Calling Conditions

Multithread safe.

Arguments

pFrontData Touch data of the front touch panel
pBackData Touch data of the back touch panel

Return Values

Value	Description
SCE_OK	Normal completion
SCE_SYSTEM_GESTURE_ERROR_INVALID_ARGUMENT	Both <i>pFrontData</i> and <i>pFrontData</i> were NULL
SCE_SYSTEM_GESTURE_ERROR_NOT_INITIALIZED	The primitive touch gesture recognizer was not initialized

Description

This function updates the primitive touch gesture recognizer. In each frame, newly acquire the touch data of the front touch panel, back touch panel, or both touch panels using libtouch, and pass this data to this function. For how to acquire touch data, refer to "Touch Service Overview" and "Touch Service Reference".

See Also

sceSystemGestureInitializePrimitiveTouchRecognizer()

sceSystemGestureGetPrimitiveTouchEvents

Get all primitive touch events

Definition

```
#include <systemgesture.h>
int sceSystemGestureGetPrimitiveTouchEvents (
    SceSystemGesturePrimitiveTouchEvent* primitiveEventBuffer,
    const SceUInt32 capacityOfBuffer,
    SceUInt32* numberOfEvent
);
```

Calling Conditions

Multithread safe.

Arguments

<i>primitiveEventBuffer</i>	Buffer for storing primitive touch event data
<i>capacityOfBuffer</i>	Number of primitive touch event data that can be stored to buffer
<i>numberOfEvent</i>	Number of primitive touch events that were acquired

Return Values

Value	Description
SCE_OK	Normal completion
SCE_SYSTEM_GESTURE_ERROR_INVALID_ARGUMENT	<i>primitiveEventBuffer</i> or <i>numberOfEvent</i> was NULL, or <i>capacityOfBuffer</i> was 0
SCE_SYSTEM_GESTURE_ERROR_NOT_INITIALIZED	The primitive touch gesture recognizer was not initialized

Description

This function gets all the primitive touch events. There are four primitive touch events, namely pressing the touch panel with a finger (Begin), maintaining contact with the touch panel with a finger (Active), releasing the touch panel by lifting one's finger off the touch panel (End), and not touching the touch panel with a finger (Inactive). This function can acquire the primitive touch events in Begin/Active/End states.

The maximum value of *numberOfEvent* is 24 (6 (maximum number of touch points that can be acquired by the touch panel) x 2 (if finger press and release was performed simultaneously) x 2 (number of touch panels)).

An execution example is given below.

```
int num = 0;
SceSystemGesturePrimitiveTouchEvent buf[24];
int ret = sceSystemGestureGetPrimitiveTouchEvents(buf, 24, &num);
for( int i=0; i<num; i++ )
{
    // Do something with the received primitive touch event buf[i]
}
```

See Also

SceSystemGesturePrimitiveTouchEvent

SCE CONFIDENTIAL

sceSystemGestureGetPrimitiveTouchEventsCount

Get number of primitive touch events in Begin/Active/End states

Definition

```
#include <systemgesture.h>
int sceSystemGestureGetPrimitiveTouchEventsCount();
```

Calling Conditions

Multithread safe.

Arguments

None

Return Values

Value	Description
0 or greater	Number of primitive touch events in Begin/Active/End states
SCE_SYSTEM_GESTURE_ERROR_NOT_INITIALIZED	The primitive touch gesture recognizer was not initialized

Description

Of the primitive touch events, this function returns the number of those that are in the Begin/Active/End state.

See Also

SceSystemGesturePrimitiveTouchEvent

SCE CONFIDENTIAL

sceSystemGestureGetPrimitiveTouchEventByIndex

Get primitive touch event specified by index

Definition

```
#include <systemgesture.h>
int sceSystemGestureGetPrimitiveTouchEventByIndex (
    const SceUInt32 index,
    SceSystemGesturePrimitiveTouchEvent* primitiveTouchEvent
);
```

Calling Conditions

Multithread safe.

Arguments

<i>index</i>	Index of primitive touch events to be acquired
<i>primitiveTouchEvent</i>	Buffer for storing primitive touch event data

Return Values

Value	Description
SCE_OK	Normal completion
SCE_SYSTEM_GESTURE_ERROR_INVALID_ARGUMENT	<i>primitiveTouchEvent</i> was NULL
SCE_SYSTEM_GESTURE_ERROR_INDEX_OUT_OF_ARRAY	Upper limit of <i>index</i> was exceeded
SCE_SYSTEM_GESTURE_ERROR_NOT_INITIALIZED	The primitive touch gesture recognizer was not initialized

Description

This function gets primitive touch events by specifying *index*. The range of values that can be assigned to the *index* is from 0 to `sceSystemGestureGetPrimitiveTouchEventsCount()` - 1. It is assumed that this function will be called as follows.

```
int ret = 0;
SceSystemGesturePrimitiveTouchEvent touchEvent;
int eventCount = sceSystemGestureGetPrimitiveTouchEventsCount();
for( int i=0; i<eventCount; i++ )
{
    ret = sceSystemGestureGetPrimitiveTouchEventByIndex(i, &touchEvent)

    // Do something with the received primitive touch event
}
```

See Also

SceSystemGesturePrimitiveTouchEvent

sceSystemGestureGetPrimitiveTouchEventByPrimitiveID

Get primitive touch event specified by event ID

Definition

```
#include <systemgesture.h>
int sceSystemGestureGetPrimitiveTouchEventByPrimitiveID (
    const SceUInt16 primitiveID,
    SceSystemGesturePrimitiveTouchEvent* primitiveTouchEvent
);
```

Calling Conditions

Multithread safe.

Arguments

primitiveID Event ID of primitive touch event
primitiveTouchEvent Buffer for storing primitive touch event data

Return Values

Value	Description
SCE_OK	Normal completion
SCE_SYSTEM_GESTURE_ERROR_INVALID_ARGUMENT	<i>primitiveTouchEvent</i> was NULL
SCE_SYSTEM_GESTURE_ERROR_EVENT_DATA_NOT_FOUND	No primitive touch event having the target <i>primitiveID</i> was found
SCE_SYSTEM_GESTURE_ERROR_NOT_INITIALIZED	The primitive touch gesture recognizer was not initialized

Description

This function gets primitive touch events by specifying the event ID. It is used to perform monitoring by tracking the states of specific primitive touch events.

The following is an example of tracking the first found Begin state item among the acquired primitive touch events using `sceSystemGestureGetPrimitiveTouchEventByIndex()`.

```
// global
int primitiveID = -1;

// in every frame
int ret = 0;
SceSystemGesturePrimitiveTouchEvent touchEvent;

if( primitiveID < 0 )
{
    int eventCount = sceSystemGestureGetPrimitiveTouchEventsCount();
    for( int i=0; primitiveID < 0 && i<eventCount; i++ )
    {
        ret = sceSystemGestureGetPrimitiveTouchEventByIndex(i,
            &touchEvent)
        if( touchEvent.eventState ==
            SCE_SYSTEM_GESTURE_TOUCH_STATE_BEGIN )
        {

```

SCE CONFIDENTIAL

```
        primitiveID = touchEvent.primitiveID;
    }
}
else
{
    ret = sceSystemGestureGetPrimitiveTouchEventByPrimitiveID(primitiveID,
&touchEvent);

    // Do something with received primitive touch event
}
```

See Also

SceSystemGesturePrimitiveTouchEvent

SCE CONFIDENTIAL

sceSystemGestureCreateTouchRecognizer

Create touch gesture recognizer

Definition

```
#include <systemgesture.h>
int sceSystemGestureCreateTouchRecognizer(
    SceSystemGestureTouchRecognizer* touchRecognizer,
    const SceSystemGestureType gestureType,
    const SceUInt8 touchPanelPortID,
    const SceSystemGestureRectangle* rectangle,
    const SceSystemGestureTouchRecognizerParameter*
    touchRecognizerParameter
);
```

Calling Conditions

Multithread safe.

Arguments

<i>touchRecognizer</i>	Buffer of touch gesture recognizer
<i>gestureType</i>	Gesture type
<i>touchPanelPortID</i>	Port number of touch panel
<i>rectangle</i>	Recognition target rectangle of touch gesture recognizer
<i>touchRecognizerParameter</i>	Parameter for initialization of touch gesture recognizer

Return Values

Value	Description
SCE_OK	Normal completion
SCE_SYSTEM_GESTURE_ERROR_INVALID_ARGUMENT	<i>touchRecognizer</i> was NULL
SCE_SYSTEM_GESTURE_ERROR_NOT_SUPPORTED_GESTURE	The gesture type specified with <i>gestureType</i> is not supported

Description

This function creates a touch gesture recognizer for gestures such as Tap or Drag.

The following values can be specified for *gestureType*. Multiple gesture recognizers can be generated for the same *gestureType*.

<i>gestureType</i>	Description
SCE_SYSTEM_GESTURE_TYPE_TAP	Specify this value to create a Tap gesture recognizer.
SCE_SYSTEM_GESTURE_TYPE_DRAG	Specify this value to create a Drag gesture recognizer.
SCE_SYSTEM_GESTURE_TYPE_TAP_AND_HOLD	Specify this value to create a Tap and Hold gesture recognizer.
SCE_SYSTEM_GESTURE_TYPE_PINCH_OUT_IN	Specify this value to create a Pinch Out/In gesture recognizer.
SCE_SYSTEM_GESTURE_TYPE_ROTATION	Specify this value to create a Rotation gesture recognizer.

For *touchPanelPortID*, specify SCE_TOUCH_PORT_FRONT or SCE_TOUCH_PORT_BACK to select the front touch panel or back touch panel, respectively, as the target panel.

©SCEI

For *rectangle*, specify in the touch panel coordinate system the rectangular area to be the recognition target of the touch gesture recognizer to be created. NULL can also be specified for *rectangle*. In this case, the entire touch panel area is the recognition target area.

For *touchRecognizerParameter*, specify the parameters for initializing the touch gesture recognizer to be created. If NULL is specified, the default parameters are set. For the parameters that can be set for the various touch gestures, refer to the structure for that parameter, such as *SceSystemGestureTapRecognizerParameter*.

Examples of creation of various touch gesture recognizers are given below.

```
SceSystemGestureTouchRecognizer tapReocognizer;
SceSystemGestureTouchRecognizer multiTapRecognizer;
SceSystemGestureTouchRecognizer dragRecognizer;
SceSystemGestureTouchRecognizer tapAndHoldRecognizer;
SceSystemGestureTouchRecognizer pinchOutInRecognizer;
SceSystemGestureTouchRecognizer rotationRecognizer;
SceSystemGestureRectangle rect;

// initialize Tap recognizer
rect.x = 0;
rect.y = 0;
rect.width = 960;
rect.height = 544;
sceSystemGestureCreateTouchRecognizer( SCE_SYSTEM_GESTURE_TYPE_TAP,
&tapReocognizer, SCE_TOUCH_PORT_FRONT, &rect, NULL );

// initialize Tap and Hold recognizer
rect.x = 960;
SceSystemGestureTouchRecognizerParameter tapAndHoldParam;
tapAndHoldParam.tapAndHold.timeToInvokeEvent = 1000;
sceSystemGestureCreateTouchRecognizer( SCE_SYSTEM_GESTURE_TYPE_TAP_AND_HOLD,
&tapAndHoldRecognizer, SCE_TOUCH_PORT_FRONT, &rect, &tapAndHoldParam );

// initialize Multi-tap recognizer
SceSystemGestureTouchRecognizerParameter multiTapParam;
multiTapParam.tap.maxTapCount = 2;
sceSystemGestureCreateTouchRecognizer( SCE_SYSTEM_GESTURE_TYPE_TAP,
&multiTapRecognizer, SCE_TOUCH_PORT_BACK, NULL, &multiTapParam );

// initialize Drag recognizer
sceSystemGestureCreateTouchRecognizer( SCE_SYSTEM_GESTURE_TYPE_DRAG,
&dragRecognizer, SCE_TOUCH_PORT_FRONT, NULL, NULL );

// initialize Pinch Out/In recognizer
rect.x = 200;
rect.y = 0;
rect.width = 1920 - 200 * 2;
rect.height = 1088;
sceSystemGestureCreateTouchRecognizer( SCE_SYSTEM_GESTURE_TYPE_PINCH_OUT_IN,
&pinchOutInRecognizer, SCE_TOUCH_PORT_BACK, &rect, NULL );

sceSystemGestureCreateTouchRecognizer( SCE_SYSTEM_GESTURE_TYPE_ROTATION,
&rotationRecognizer, SCE_TOUCH_PORT_BACK, &rect, NULL );
```

See Also

SceSystemGestureTouchRecognizer, *SceSystemGestureRectangle*,
SceSystemGestureTouchRecognizerParameter

SCE CONFIDENTIAL

sceSystemGestureResetTouchRecognizer

Reset touch gesture recognizer

Definition

```
#include <systemgesture.h>
int sceSystemGestureResetTouchRecognizer(
    SceSystemGestureTouchRecognizer* touchRecognizer
);
```

Calling Conditions

Multithread safe.

Arguments

touchRecognizer Buffer of touch gesture recognizer

Return Values

Value	Description
SCE_OK	Normal completion
SCE_SYSTEM_GESTURE_ERROR_INVALID_ARGUMENT	<i>touchRecognizer</i> was NULL
SCE_SYSTEM_GESTURE_ERROR_NOT_INITIALIZED	The touch gesture recognizer was not initialized

Description

This function resets the touch gesture recognizer. All the touch events related to the touch gesture recognizer specified with *touchRecognizer* are returned to their initial state.

See Also

SceSystemGestureTouchRecognizer

sceSystemGestureGetTouchRecognizerInformation

Get information of touch gesture recognizer

Definition

```
#include <systemgesture.h>
int sceSystemGestureGetTouchRecognizerInformation (
    const SceSystemGestureTouchRecognizer* touchRecognizer,
    SceSystemGestureTouchRecognizerInformation* information
);
```

Calling Conditions

Multithread safe.

Arguments

<i>touchRecognizer</i>	Buffer of touch gesture recognizer
<i>information</i>	Information of touch gesture recognizer

Return Values

Value	Description
SCE_OK	Normal completion
SCE_SYSTEM_GESTURE_ERROR_INVALID_ARGUMENT	<i>touchRecognizer</i> was NULL
SCE_SYSTEM_GESTURE_ERROR_NOT_INITIALIZED	The touch gesture recognizer was not initialized

Description

This function gets the information of the touch gesture recognizer. The information related to the touch gesture recognizer specified with *touchRecognizer* is stored to *information*. For the information that can be acquired, refer to the *SceSystemGestureTouchRecognizerInformation* section.

See Also

SceSystemGestureTouchRecognizer, *SceSystemGestureTouchRecognizerInformation*

SCE CONFIDENTIAL

sceSystemGestureUpdateTouchRecognizer

Update touch gesture recognizer

Definition

```
#include <systemgesture.h>
int sceSystemGestureUpdateTouchRecognizer (
    SceSystemGestureTouchRecognizer* touchRecognizer
);
```

Calling Conditions

Multithread safe.

Arguments

touchRecognizer Buffer of touch gesture recognizer

Return Values

Value	Description
SCE_OK	Normal completion
SCE_SYSTEM_GESTURE_ERROR_INVALID_ARGUMENT	<i>touchRecognizer</i> was NULL
SCE_SYSTEM_GESTURE_ERROR_NOT_INITIALIZED	The touch gesture recognizer was not initialized

Description

This function updates the touch gesture recognizer. Make sure to always call this function for each gesture recognizer after the primitive touch gesture recognizers have been updated using `sceSystemGestureUpdatePrimitiveTouchRecognizer()` in each frame.

See Also

`SceSystemGestureTouchRecognizer`,
`sceSystemGestureUpdatePrimitiveTouchRecognizer()`

SCE CONFIDENTIAL

sceSystemGestureUpdateTouchRecognizerRectangle

Update recognition target rectangle of touch gesture recognizer

Definition

```
#include <systemgesture.h>
int sceSystemGestureUpdateTouchRecognizerRectangle (
    SceSystemGestureTouchRecognizer* touchRecognizer,
    const SceSystemGestureRectangle* rectangle
);
```

Calling Conditions

Multithread safe.

Arguments

<i>touchRecognizer</i>	Buffer of touch gesture recognizer
<i>rectangle</i>	Recognition target rectangle of touch gesture recognizer

Return Values

Value	Description
SCE_OK	Normal completion
SCE_SYSTEM_GESTURE_ERROR_INVALID_ARGUMENT	<i>touchRecognizer</i> or <i>rectangle</i> was NULL
SCE_SYSTEM_GESTURE_ERROR_NOT_INITIALIZED	The touch gesture recognizer was not initialized

Description

This function updates the recognition target rectangle area of the gesture recognizer. Call this function when the recognition target area moves or to have a different area be used as the recognition target area. To have a completely different area be used as the recognition target area, it is recommended to call `sceSystemGestureResetTouchRecognizer()` along with this function.

See Also

`SceSystemGestureTouchRecognizer`, `SceSystemGestureRectangle`,
`sceSystemGestureResetTouchRecognizer()`

sceSystemGestureGetTouchEvents

Get all touch events that were recognized by touch gesture recognizer

Definition

```
#include <systemgesture.h>
int sceSystemGestureGetTouchEvents (
    const SceSystemGestureTouchRecognizer* touchRecognizer,
    SceSystemGestureTouchEvent* eventBuffer,
    const SceUInt32 capacityOfBuffer,
    SceUInt32* numberOfEvent
);
```

Calling Conditions

Multithread safe.

Arguments

<i>touchRecognizer</i>	Buffer of touch gesture recognizer
<i>eventBuffer</i>	Buffer for storing touch event data
<i>capacityOfBuffer</i>	Number of touch event data that can be stored to buffer
<i>numberOfEvent</i>	Number of touch events that were acquired

Return Values

Value	Description
SCE_OK	Normal completion
SCE_SYSTEM_GESTURE_ERROR_INVALID_ARGUMENT	<i>touchRecognizer</i> , <i>eventBuffer</i> or <i>numberOfEvent</i> was NULL, or <i>capacityOfBuffer</i> was 0
SCE_SYSTEM_GESTURE_ERROR_NOT_INITIALIZED	The touch gesture recognizer was not initialized

Description

This function gets all the touch events of the touch gesture recognizer.

The maximum value of *numberOfEvent* is 6 (maximum number of touch points that can be acquired by the touch panel).

An execution example is given below.

```
int r = 0;
int num = 0;
SceSystemGestureTouchEvent buf[6];
r = sceSystemGestureGetTouchEvents(&touchRecognizer, buf, sizeof(buf),
&num);
for( int i=0; i<num; i++ )
{
    // Do something with the received touch event buf[i]
}
```

See Also

SceSystemGestureTouchRecognizer, SceSystemGestureTouchEvent

SCE CONFIDENTIAL

sceSystemGestureGetTouchEventsCount

Get number of touch events that were recognized by touch gesture recognizer

Definition

```
#include <systemgesture.h>
int sceSystemGestureGetTouchEventsCount (
    const SceSystemGestureTouchRecognizer* touchRecognizer
);
```

Calling Conditions

Multithread safe.

Arguments

touchRecognizer Buffer of touch gesture recognizer

Return Values

Value	Description
0 or greater	Number of touch events that were recognized by touch gesture recognizer
SCE_SYSTEM_GESTURE_ERROR_INVALID_ARGUMENT	<i>touchRecognizer</i> was NULL
SCE_SYSTEM_GESTURE_ERROR_NOT_INITIALIZED	The touch gesture recognizer was not initialized

Description

This function returns the number of touch events that were recognized by the touch gesture recognizer.

See Also

SceSystemGestureTouchRecognizer

SCE CONFIDENTIAL

sceSystemGestureGetTouchEventByIndex

Get touch event specified by index

Definition

```
#include <systemgesture.h>
int sceSystemGestureGetTouchEventByIndex (
    const SceSystemGestureTouchRecognizer* touchRecognizer,
    const SceUInt32 index,
    SceSystemGestureTouchEvent* touchEvent
);
```

Calling Conditions

Multithread safe.

Arguments

<i>touchRecognizer</i>	Buffer of touch gesture recognizer
<i>index</i>	Index of touch event to be acquired
<i>touchEvent</i>	Buffer for storing touch event data

Return Values

Value	Description
SCE_OK	Normal completion
SCE_SYSTEM_GESTURE_ERROR_INVALID_ARGUMENT	<i>touchRecognizer</i> or <i>touchEvent</i> was NULL
SCE_SYSTEM_GESTURE_ERROR_NOT_INITIALIZED	The touch gesture recognizer was not initialized
SCE_SYSTEM_GESTURE_ERROR_INDEX_OUT_OF_ARRAY	Upper limit of <i>index</i> was exceeded

Description

This function gets touch events by specifying *index*. The range of values that can be assigned to *index* is from 0 to `sceSystemGestureGetTouchEventsCount()` - 1. It is assumed that this function will be called as follows.

```
int r = 0;
SceSystemGestureTouchEvent touchEvent;
int eventCount = sceSystemGestureGetTouchEventsCount(&touchRecognizer);
for( int i=0; i<eventCount; i++ )
{
    r = sceSystemGestureGetTouchEventByIndex(&touchRecognizer, i,
    &touchEvent)

    // Do something with the received touch event
}
```

See Also

`SceSystemGestureTouchRecognizer`, `SceSystemGestureTouchEvent`

sceSystemGestureGetTouchEventByEventID

Get touch event specified by event ID

Definition

```
#include <systemgesture.h>
int sceSystemGestureGetTouchEventByEventID (
    const SceSystemGestureTouchRecognizer* touchRecognizer,
    const SceUInt32 eventID,
    SceSystemGestureTouchEvent* touchEvent
);
```

Calling Conditions

Multithread safe.

Arguments

<i>touchRecognizer</i>	Buffer of touch gesture recognizer
<i>eventID</i>	Event ID of touch event
<i>touchEvent</i>	Buffer for storing touch event data

Return Values

Value	Description
SCE_OK	Normal completion
SCE_SYSTEM_GESTURE_ERROR_INVALID_ARGUMENT	<i>touchRecognizer</i> or <i>touchEvent</i> was NULL
SCE_SYSTEM_GESTURE_ERROR_NOT_INITIALIZED	The touch gesture recognizer was not initialized
SCE_SYSTEM_GESTURE_ERROR_EVENT_DATA_NOT_FOUND	No touch event having the target <i>eventID</i> was found

Description

This function gets touch events by specifying the event ID. It is used to perform monitoring by tracking the states of specific touch events.

The following is an example of tracking the first found Begin state item among the acquired Drag gesture events using `sceSystemGestureGetTouchEventByIndex()`.

```
// global
int dragEventID = 0;

// in every frame
int r = 0;
SceSystemGestureTouchEvent dragEvent;

if( dragEventID == 0 )
{
    int eventCount =
    sceSystemGestureGetTouchEventsCount(&dragRecognizer);
    for( int i=0; dragEventID == 0 && i<eventCount; i++ )
    {
        r = sceSystemGestureGetTouchEventByIndex(&dragRecognizer, i,
        &dragEvent)
        dragEventID = dragEvent.eventID;
    }
}
```

SCE CONFIDENTIAL

```
        }  
    }  
    else  
    {  
        r = sceSystemGestureGetTouchEventByEventID(&dragRecognizer, dragEventID,  
            &dragEvent);  
  
        // Do something with received primitive touch event  
    }
```

See Also

SceSystemGestureTouchRecognizer, SceSystemGestureTouchEvent

000004892117