Class

AVCaptureEventInteraction

An object that registers handlers to respond to capture events from system hardware buttons.

iOS 17.2+ | iPadOS 17.2+ | Mac Catalyst 17.2+

```
@MainActor
class AVCaptureEventInteraction
```

Overview

The system Camera app allows people to perform capture functions by pressing hardware buttons on their iOS device. UIKit apps can add similar functionality by using this type to register handlers that respond to interactions from device hardware.

> **Note**
>
> In SwiftUI, respond to capture events from hardware buttons using `onCameraCaptureEvent(isEnabled:action:)` and `onCameraCaptureEvent(isEnabled:primaryAction:secondaryAction:)` instead.

The following example shows how to add a handler that captures a photo when a user presses a hardware button on their device.

```
class CameraViewController: UIViewController {

    /// An object that manages the camera functionality.
    private let camera = CameraModel()
    /// A capture event interaction to handle hardware button presses.
    private var eventInteraction: AVCaptureEventInteraction?

    override func viewDidLoad() {
        super.viewDidLoad()
        // Configure the app to take a photo on hardware button press.
        configureHardwareInteraction()
    }

    private func configureHardwareInteraction() {
        // Create a new capture event interaction with a handler that captures a pho
        let interaction = AVCaptureEventInteraction { [weak self] event in
            // Capture a photo on "press up" of a hardware button.
            if event.phase == .ended {
                self?.camera.capturePhoto()
            }
        }
        // Add the interaction to the view controller's view.
        view.addInteraction(interaction)
        eventInteraction = interaction
    }
}
```

The event handler queries the capture event to determine its phase, and when the interaction ends, captures a photo.

> **Important**
>
> You can only use this API for capture use cases. The system sends capture events only to apps that actively use the camera. Backgrounded capture apps, and apps not performing capture, don’t receive events.
>
> Adopting this API overrides default hardware button behavior, so apps must always respond appropriately to any events received. Failing to handle events results in a nonfunctional button that provides a poor user experience. If your app is temporarily unable to handle events, disable the interaction by setting its `isEnabled` property to `false`, which restores the system button behavior.

Topics

Creating an interaction

`init(handler: (AVCaptureEvent) -> Void)`

Creates a capture event interaction with a handler that responds to presses of hardware buttons.

`init(primary: (AVCaptureEvent) -> Void, secondary: (AVCaptureEvent) -> Void)`

Creates a capture event interaction with handlers that respond independently to presses of hardware buttons.

Inspecting the interaction

`var isEnabled: Bool`

A Boolean value that indicates whether this capture event interaction is in an enabled state.

`class var defaultCaptureSoundDisabled: Bool`

A Boolean value that indicates whether the default sound is in a disabled state.

Relationships

Inherits From

`NSObject`

Conforms To

```
CVarArg
CustomDebugStringConvertible
CustomStringConvertible
Equatable
Hashable
NSObjectProtocol
Sendable
UIInteraction
```

See Also

iOS playback and capture

{} Playing video content in a standard user interface

Play media full screen, embedded inline, or in a floating Picture in Picture (PiP) window using a player view controller.

`class AVPlayerViewController`

A view controller that displays content from a player and presents a native user interface to control playback.

`protocol AVPlayerViewControllerDelegate`

A protocol that defines the methods to implement to respond to player view controller events.

`class AVCaptureEvent`

An object that describes a user interaction with a system hardware button.

`class AVCaptureEventSound`

A sound object for a capture event.

`class AVInputPickerInteraction`

Use `AVInputPickerInteraction` to present an input picker.