Class

TrackOpticalFlowRequest

A request that determines the direction change of vectors for each pixel from a previous to current image.

iOS 18.0+ | iPadOS 18.0+ | macOS 15.0+ | tvOS 18.0+ | visionOS 2.0+

```
final class TrackOpticalFlowRequest
```

Overview

This request generates an `OpticalFlowObservation` object that describes the directional change from image to image. The request works at the pixel level, so both images needs to have the same dimensions to successfully perform the request.

Important

Optical flow requests are very resource intensive, so perform only one request at a time. Release memory immediately after generating an optical flow.

Topics

Creating a request

`init(TrackOpticalFlowRequest.Revision?, frameAnalysisSpacing: CMTime?)`

Creates an optical-flow tracking request.

Getting the revision

```
let revision: TrackOpticalFlowRequest.Revision
```

The algorithm or implementation the request uses.

```
static let supportedRevisions: [TrackOpticalFlowRequest.Revision]
```

The collection of revisions the request supports.

```
enum Revision
```

A type that describes the algorithm or implementation that the request performs.

Inspecting a request

```
var computationAccuracy: TrackOpticalFlowRequest.ComputationAccuracy
```

The level of accuracy to compute the optical flow.

```
enum ComputationAccuracy
```

A type that describes the computational accuracy.

```
var outputPixelFormatType: OSType
```

The desired pixel format type of the observation.

```
var supportedOutputPixelFormatTypes: [OSType]
```

The collection of supported pixel format types.

Performing a request

```
func perform(on: URL, orientation: CGImagePropertyOrientation?) async
throws -> Self.Result
```

Performs the request on an image URL and produces observations.

Required Default implementations provided.

```
func perform(on: Data, orientation: CGImagePropertyOrientation?) async
throws -> Self.Result
```

Performs the request on image data and produces observations.

Required Default implementations provided.

```
func perform(on: CGImage, orientation: CGImagePropertyOrientation?)
async throws -> Self.Result
```

Performs the request on a Core Graphics image and produces observations.

Required Default implementations provided.

```
func perform(on: CVPixelBuffer, orientation: CGImagePropertyOrientation
?) async throws -> Self.Result
```

Performs the request on a pixel buffer and produces observations.

Required Default implementations provided.

```
func perform(on: CMSampleBuffer, orientation: CGImageProperty
Orientation?) async throws -> Self.Result
```

Performs the request on a Core Media buffer and produces observations.

Required Default implementations provided.

```
func perform(on: CIImage, orientation: CGImagePropertyOrientation?)
async throws -> Self.Result
```

Performs the request on a Core Image image and produces observations.

Required Default implementations provided.

```
struct OpticalFlowObservation
```

An object that represents an optical flow that an image-analysis request produces.

Relationships

Conforms To

`CustomStringConvertible`
`Equatable`
`Hashable`
`ImageProcessingRequest`
`Sendable`
`SendableMetatype`
`StatefulRequest`
`TargetedRequest`
`VisionRequest`

See Also

Optical flow and rectangle detection

```
struct DetectRectanglesRequest
```

An image-analysis request that finds projected rectangular regions in an image.