CIVector Class Reference

Cocoa > Graphics & Imaging

2006-12-07

Apple Inc.
© 2006 Apple Computer, Inc.
All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, electronic, photocopying, recording, or otherwise, without prior written permission of Apple Inc., with the following exceptions: Any person is hereby authorized to store documentation on a single computer for personal use only and to print copies of documentation for personal use provided that the documentation contains Apple's copyright notice.

The Apple logo is a trademark of Apple Inc.

Use of the "keyboard" Apple logo (Option-Shift-K) for commercial purposes without the prior written consent of Apple may constitute trademark infringement and unfair competition in violation of federal and state laws.

No licenses, express or implied, are granted with respect to any of the technology described in this document. Apple retains all intellectual property rights associated with the technology described in this document. This document is intended to assist application developers to develop applications only for Apple-labeled computers.

Every effort has been made to ensure that the information in this document is accurate. Apple is not responsible for typographical errors.

Apple Inc.
1 Infinite Loop
Cupertino, CA 95014
408-996-1010

.Mac is a registered service mark of Apple Inc.

Apple, the Apple logo, Carbon, Cocoa, Mac, Mac OS, Quartz, and QuickTime are trademarks of Apple Inc., registered in the United States and other countries.

Simultaneously published in the United States and Canada.

Even though Apple has reviewed this document, APPLE MAKES NO WARRANTY OR REPRESENTATION, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT, ITS QUALITY, ACCURACY, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. AS A RESULT, THIS DOCUMENT IS PROVIDED "AS IS," AND YOU, THE READER, ARE ASSUMING THE ENTIRE RISK AS TO ITS QUALITY AND ACCURACY.

IN NO EVENT WILL APPLE BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY DEFECT OR INACCURACY IN THIS DOCUMENT, even if advised of the possibility of such damages.

THE WARRANTY AND REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESS OR IMPLIED. No Apple dealer, agent, or employee is authorized to make any modification, extension, or addition to this warranty.

Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

Contents

CIVector Class Reference

Inherits from	NSObject
Conforms to	NSCoding NSCopying NSObject (NSObject)
Framework	Library/Frameworks/QuartzCore.framework
Availability	Mac OS X v10.4 and later
Companion guide	Core Image Programming Guide
Declared in	CIVector.h
Related sample code	CarbonCocoaCoreImageTab CIAnnotation CITransitionSelectorSample2 CIVideoDemoGL Reducer

Overview

The `CIVector` class is used for coordinate values and direction vectors. You typically use a `CIVector` object to pass parameter values to Core Image filters. `CIVector` objects work in conjunction with other Core Image classes, such as `CIFilter`, `CIContext`, `CIImage`, and `CIColor`, to process images using the Core Image framework.

Tasks

Creating a Vector

`+ vectorWithValues:count:` (page 7)
Creates and returns a vector that is initialized with the specified values.

`+ vectorWithX:` (page 8)
Creates and returns a vector that is initialized with one value.

`+ vectorWithX:Y:` (page 8)
Creates and returns a vector that is initialized with two values.

+ `vectorWithX:Y:Z:` (page 8)
  Creates and returns a vector that is initialized with three values.

+ `vectorWithX:Y:Z:W:` (page 9)
  Creates and returns a vector that is initialized with four values.

+ `vectorWithString:` (page 7)
  Creates and returns a vector that is initialized with values provided in a string representation.

Initializing a Vector

– `initWithValues:count:` (page 10)
  Initializes a vector with the provided values.

– `initWithX:` (page 11)
  Initializes the first position of a vector with the provided values.

– `initWithX:Y:` (page 11)
  Initializes the first two positions of a vector with the provided values.

– `initWithX:Y:Z:` (page 11)
  Initializes the first three positions of a vector with the provided values.

– `initWithX:Y:Z:W:` (page 12)
  Initializes four positions of a vector with the provided values.

– `initWithString:` (page 10)
  Initializes a vector with values provided in a string representation.

Getting Values From a Vector

– `valueAtIndex:` (page 13)
  Returns a value from a specific position in a vector.

– `count` (page 10)
  Returns the number of items in a vector.

– `X` (page 13)
  Returns the value located in the first position in a vector.

– `Y` (page 14)
  Returns the value located in the second position in a vector.

– `Z` (page 14)
  Returns the value located in the third position in a vector.

– `W` (page 13)
  Returns the value located in the fourth position in a vector.

– `stringRepresentation` (page 12)
  Returns a string representation for a vector.

Class Methods

vectorWithString:

Creates and returns a vector that is initialized with values provided in a string representation.

```
+ (CIVector *)vectorWithString:(NSString *)representation
```

Parameters

representation

A string that is in one of the formats returned by the `stringRepresentation` method.

Discussion

Some typical string representations for vectors are:

```
@"[1.0 0.5 0.3]"
```

which specifies a `vec3` vector whose components are `X = 1.0`, `Y = 0.5`, and `Z = 0.3`

```
@"[10.0 23.0]
```

which specifies a `vec2` vector show components are `X = 10.0` and `Y = 23.0`

Availability

Mac OS X v10.4 and later.

See Also

`- stringRepresentation` (page 12)

Declared In

`CIVector.h`

vectorWithValues:count:

Creates and returns a vector that is initialized with the specified values.

```
+ (CIVector *)vectorWithValues:(const CGFloat *)values count:(size_t)count
```

Parameters

values

The values to initialize the vector with.

count

The number of values in the vector.

Return Value

A vector initialized with the provided values.

Availability

Mac OS X v10.4 and later.

Declared In

`CIVector.h`

vectorWithX:

Creates and returns a vector that is initialized with one value.

```
+ (CIVector *)vectorWithX:(CGFloat)x
```

Parameters

x

The value to initialize the vector with.

Return Value

A vector initialized with the specified value.

Availability

Mac OS X v10.4 and later.

Declared In

`CIVector.h`

vectorWithX:Y:

Creates and returns a vector that is initialized with two values.

```
+ (CIVector *)vectorWithX:(CGFloat)x Y:(CGFloat)y
```

Parameters

x

The value for the first position in the vector.

y

The value for the second position in the vector.

Return Value

A vector initialized with the specified values.

Availability

Mac OS X v10.4 and later.

Related Sample Code

CarbonCocoaCoreImageTab

CIAnnotation

CIVideoDemoGL

Core Animation QuickTime Layer

Reducer

Declared In

`CIVector.h`

vectorWithX:Y:Z:

Creates and returns a vector that is initialized with three values.

```
+ (CIVector *)vectorWithX:(CGFloat)x Y:(CGFloat)y Z:(CGFloat)z
```

Parameters

x

The value for the first position in the vector.

y

The value for the second position in the vector.

z

The value for the third position in the vector.

Return Value

A vector initialized with the specified values.

Availability

Mac OS X v10.4 and later.

Declared In

`CIVector.h`

vectorWithX:Y:Z:W:

Creates and returns a vector that is initialized with four values.

```
+ (CIVector *)vectorWithX:(CGFloat)x Y:(CGFloat)y Z:(CGFloat)z W:(CGFloat)w
```

Parameters

x

The value for the first position in the vector.

y

The value for the second position in the vector.

z

The value for the third position in the vector.

w

The value for the fourth position in the vector.

Return Value

A vector initialized with the specified values.

Availability

Mac OS X v10.4 and later.

Related Sample Code

CarbonCocoaCoreImageTab

CIAnnotation

CITransitionSelectorSample2

Reducer

Declared In

`CIVector.h`

Instance Methods

count

Returns the number of items in a vector.

```
- (size_t)count
```

Return Value

The number of items in the vector.

Availability

Mac OS X v10.4 and later.

Declared In

`CIVector.h`

initWithString:

Initializes a vector with values provided in a string representation.

```
- (id)initWithString:(NSString *)representation;
```

Parameters

representation

A string that is in one of the formats returned by the `stringRepresentation` method.

Availability

Mac OS X v10.4 and later.

See Also

- `stringRepresentation` (page 12)

Declared In

`CIVector.h`

initWithValues:count:

Initializes a vector with the provided values.

```
- (id)initWithValues:(const CGFloat *)values count:(size_t)count
```

Parameters

values

The values to initialize the vector with.

count

The number of values specified by the `values` argument.

Availability

Mac OS X v10.4 and later.

Declared In
`CIVector.h`

initWithX:

Initializes the first position of a vector with the provided values.

```
- (id)initWithX:(CGFloat)x
```

Parameters

x

The initialization value.

Availability
Mac OS X v10.4 and later.

Declared In
`CIVector.h`

initWithX:Y:

Initializes the first two positions of a vector with the provided values.

```
- (id)initWithX:(CGFloat)x Y:(CGFloat)y
```

Parameters

x

The initialization value for the first position.

y

The initialization value for the second position.

Availability
Mac OS X v10.4 and later.

Declared In
`CIVector.h`

initWithX:Y:Z:

Initializes the first three positions of a vector with the provided values.

```
- (id)initWithX:(CGFloat)x Y:(CGFloat)y Z:(CGFloat)z
```

Parameters

x

The initialization value for the first position.

y

The initialization value for the second position.

z

The initialization value for the third position.

Availability
Available in Mac OS X v10.4 and later.

Declared In
`CIVector.h`

initWithX:Y:Z:W:

Initializes four positions of a vector with the provided values.

```
- (id)initWithX:(CGFloat)x Y:(CGFloat)y Z:(CGFloat)z W:(CGFloat)w
```

Parameters

x

The initialization value for the first position.

y

The initialization value for the second position.

z

The initialization value for the third position.

w

The initialization value for the fourth position.

Availability
Mac OS X v10.4 and later.

Declared In
`CIVector.h`

stringRepresentation

Returns a string representation for a vector.

```
- (NSString *)stringRepresentation
```

Return Value
A string object.

Discussion
You convert the string representation returned by this method to a vector by supplying it as a parameter to the `vectorWithString:` method.

Some typical string representations for vectors are:

```
@"[1.0 0.5 0.3]"
```

which specifies a `vec3` vector whose components are `X = 1.0`, `Y = 0.5`, and `Z = 0.3`

```
@"[10.0 23.0]
```

which specifies a `vec2` vector show components are `X = 10.0` and `Y = 23.0`

Availability
Mac OS X v10.4 and later.

See Also
`+ vectorWithString:` (page 7)

Declared In
`CIVector.h`

valueAtIndex:

Returns a value from a specific position in a vector.

```
- (CGFloat)valueAtIndex:(size_t)index
```

Parameters

index

The position in the vector of the value that you want to retrieve.

Return Value
The value retrieved from the vector or `0` if the position is undefined.

Discussion
The numbering of elements in a vector begins with zero.

Availability
Mac OS X v10.4 and later.

Declared In
`CIVector.h`

W

Returns the value located in the fourth position in a vector.

```
- (CGFloat)W
```

Return Value
The value retrieved from the vector.

Availability
Mac OS X v10.4 and later.

Declared In
`CIVector.h`

X

Returns the value located in the first position in a vector.

```
- (CGFloat)X
```

Return Value
The value retrieved from the vector.

Availability
Mac OS X v10.4 and later.

Related Sample Code
CIAnnotation

Declared In
`CIVector.h`

Y

Returns the value located in the second position in a vector.

```
- (CGFloat)Y
```

Return Value
The value retrieved from the vector.

Availability
Mac OS X v10.4 and later.

Related Sample Code
CIAnnotation

Declared In
`CIVector.h`

Z

Returns the value located in the third position in a vector.

```
- (CGFloat)Z
```

Return Value
The value retrieved from the vector.

Availability
Mac OS X v10.4 and later.

Declared In
`CIVector.h`

Document Revision History

This table describes the changes to *CIVector Class Reference*.

Date	Notes
2006-12-07	Updated formatting.
2006-05-23	First publication of this content as a separate document.
	Added parameter descriptions and updated Class Description.

Index