Accelerate / Creating a Core Graphics Image from a vImage Buffer

Article

Creating a Core Graphics Image from a vImage Buffer

Create displayable representations of vImage buffers.

Overview

vImage provides a function for creating Core Graphics images from vImage buffers. This function allows you to display the results of a vImage operation to your user.

Create the Image

You create a Core Graphics image from the buffer, and initialize a `UIImage` instance from that. The `createCGImage(format:flags:)` function returns a `CGImage` instance based on the supplied Core Graphics image format (for more information, see Converting bitmap data between Core Graphics images and vImage buffers).

The following example shows how to create a Core Graphics image from a vImage buffer:

```
let result = try? destinationBuffer.createCGImage(format: format)

if let result = result {
    // Assumes `imageView` is a `UIImageView`
    imageView.image = UIImage(cgImage: result)
}
```

See Also

Image Processing Essentials

Converting bitmap data between Core Graphics images and vImage buffers

Pass image data between Core Graphics and vImage to create and manipulate images.

Creating and Populating Buffers from Core Graphics Images

Initialize vImage buffers from Core Graphics images.

Building a Basic Image-Processing Workflow

Resize an image with vImage.

Applying geometric transforms to images

Reflect, shear, rotate, and scale image buffers using vImage.

Compositing images with alpha blending

Combine two images by using alpha blending to create a single output.

Compositing images with vImage blend modes

Combine two images by using blend modes to create a single output.

Applying vImage operations to regions of interest

Limit the effect of vImage operations to rectangular regions of interest.

Optimizing image-processing performance

Improve your app’s performance by converting image buffer formats from interleaved to planar.

vImage

Manipulate large images using the CPU’s vector processor.