Documentation Language: Swift

Accelerate / ··· / vImage.PixelBuffer / specifyHistogram(_:destination:)

Instance Method

specifyHistogram(_:destination:)

Performs a histogram specification operation on an 8-bit-per-channel, 3-channel multiple-plane pixel buffer.

iOS 16.0+ | iPadOS 16.0+ | Mac Catalyst | macOS 13.0+ | tvOS 16.0+ | visionOS | watchOS 9.0+

```
func specifyHistogram(
    _ histogram: vImage.PixelBuffer<Format>.Histogram888,
    destination: vImage.PixelBuffer<Format>
)
```

Available when `Format` is `vImage.Planar8x3`.

Parameters

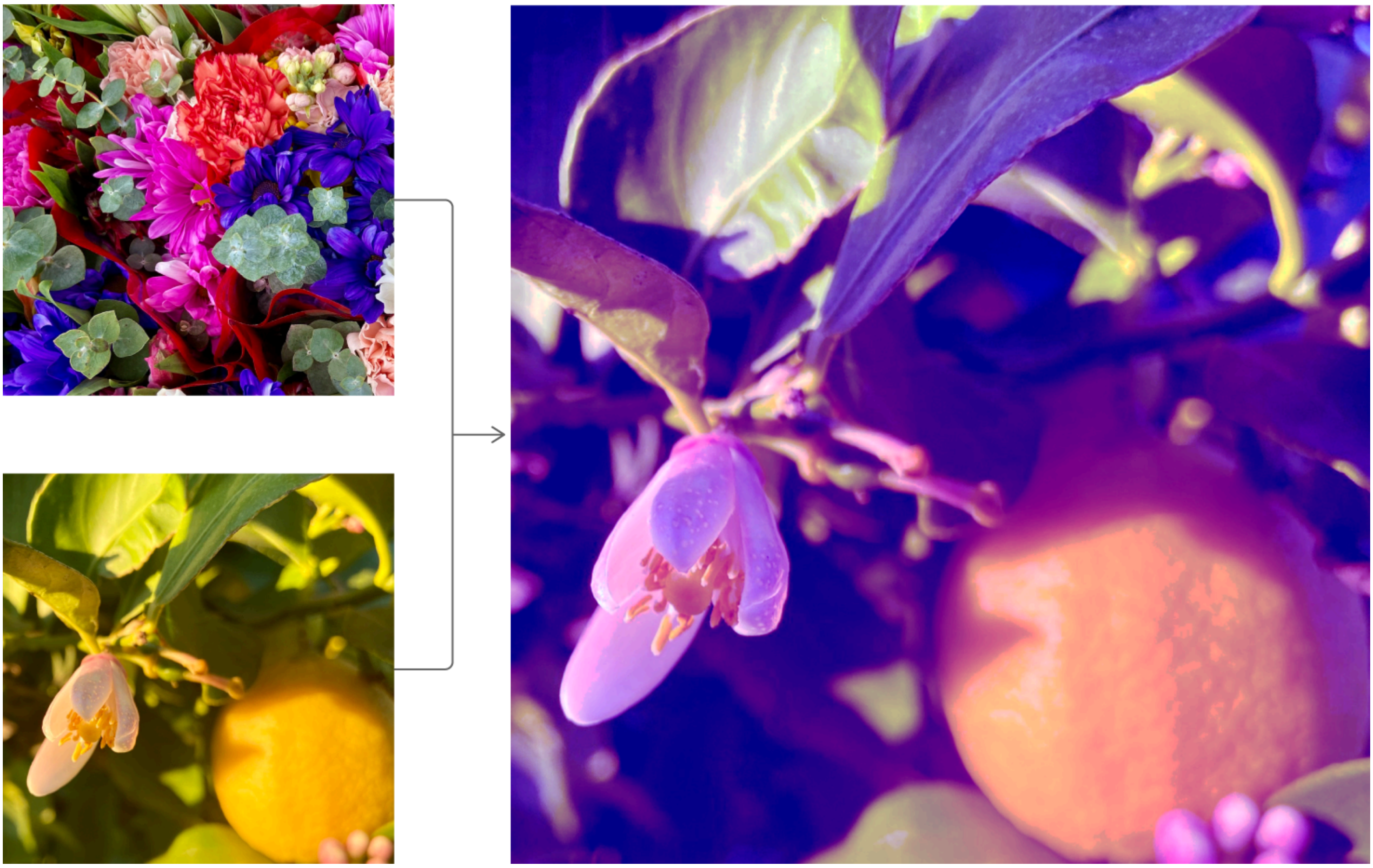
`histogram`
The histogram.

`destination`
The destination pixel buffer.

Discussion

Histogram specification is a technique that allows you to calculate the histogram of a reference image and apply it to an input image.

The example below shows a source image (bottom left) and a histogram reference image (top left), with the histogram specification output on the right:



The code below initializes the source and reference buffers from the CGImage instances `sourceImage` and `referenceImage` respectively:

```
var cgImageFormat = vImage_CGImageFormat(
    bitsPerComponent: 8,
    bitsPerPixel: 8 * 3,
    colorSpace: CGColorSpaceCreateDeviceRGB(),
    bitmapInfo: CGBitmapInfo(rawValue: CGImageAlphaInfo.none.rawValue))!

let interleavedSource = try vImage.PixelBuffer(
    cgImage: sourceImage,
    cgImageFormat: &cgImageFormat,
    pixelFormat: vImage.Interleaved8x3.self)
let multiplePlaneSource = vImage.PixelBuffer<vImage.Planar8x3>(
    planarBuffers: interleavedSource.planarBuffers())

let interleavedReference = try vImage.PixelBuffer(
    cgImage: referenceImage,
    cgImageFormat: &cgImageFormat,
    pixelFormat: vImage.Interleaved8x3.self)
let multiplePlaneReference = vImage.PixelBuffer<vImage.Planar8x3>(
    planarBuffers: interleavedReference.planarBuffers())
```

The following code calls `histogram()` to calculate the histogram of the reference image and passes the histogram to `specifyHistogram(_:destination:)` to generate the specification result. This function works in place, that is, the input and output can share the same underlying memory.

```
let histogram = multiplePlaneReference.histogram()

multiplePlaneSource.specifyHistogram(histogram,
                                     destination: multiplePlaneSource)
```

See Also

Related Documentation

{} Specifying histograms with vImage

Calculate the histogram of one image, and apply it to a second image.

Histogram specification

`func specifyHistogram(vImage.PixelBuffer<Format>.Histogram8888, destination: vImage.PixelBuffer<Format>)`

Performs a histogram specification operation on an 8-bit-per-channel, 4-channel interleaved pixel buffer.

`func specifyHistogram(vImage.PixelBuffer<Format>.HistogramFFFF, destination: vImage.PixelBuffer<Format>)`

Performs a histogram specification operation on a 32-bit-per-channel, 4-channel interleaved pixel buffer.

`func specifyHistogram(vImage.PixelBuffer<Format>.HistogramFFF, destination: vImage.PixelBuffer<Format>)`

Performs a histogram specification operation on a 32-bit-per-channel, 3-channel multiple-plane pixel buffer.

`func specifyHistogram(vImage.PixelBuffer<Format>.Histogram8888, destination: vImage.PixelBuffer<Format>)`

Performs a histogram specification operation on an 8-bit-per-channel, 4-channel multiple-plane pixel buffer.

```
func specifyHistogram(vImage.PixelBuffer<Format>.HistogramFFFF,
destination: vImage.PixelBuffer<Format>)
```

Performs a histogram specification operation on a 32-bit-per-channel, 3-channel multiple-plane pixel buffer.