Instance Method

applyPolynomial(coefficientSegments:boundaries:destination:)

Applies a set of piecewise polynomials to a 3-channel, 32-bit interleaved buffer.

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

```
func applyPolynomial(
    coefficientSegments: [[Float]],
    boundaries: [Float],
    destination: vImage.PixelBuffer<Format>
)
```

Available when `Format` is `vImage.InterleavedFx3`.

Parameters

`coefficientSegments`
An array that contains the polynomial coefficient array. Each polynomial must be of the same order.

`boundaries`
An array of boundary values, in increasing order, that separates adjacent ranges of pixel values. `boundaries` must contain `coefficientSegments.count + 1` elements.

`destination`
The destination pixel buffer.

Discussion

The following code shows an example of applying three polynomials to an `vImage.InterleavedFx3` buffer:

```
let src = vImage.PixelBuffer<vImage.InterleavedFx3>(
    pixelValues: [0.25, 0.5, 0.75],
    size: vImage.Size(width: 1, height: 1))

let dest = vImage.PixelBuffer<vImage.InterleavedFx3>(
    size: src.size)

src.applyPolynomial(coefficientSegments: [ [1, 0, 0],
                                           [0, 1, 0],
                                           [0, 0, 1] ],
                    boundaries: [0, 1/3, 2/3, 1] as [Float],
                    destination: dest)

// Prints:
//  1.0     ≅ 1 * 0.25⁰

//  0.5     ≅ 1 * 0.5¹

//  0.5625  ≅ 1 * 0.75²
print(dest.array)
```

See Also

Related Documentation

{} Applying tone curve adjustments to images

Use the vImage library's polynomial transform to apply tone curve adjustments to images.

Appying polynomial (32-bit)

`func applyPolynomial(coefficientSegments: [[Float]], boundaries: [Float], destination: vImage.PixelBuffer<Format>)`

Applies a set of piecewise polynomials to a 32-bit planar buffer.

`func applyPolynomial(coefficientSegments: [[Float]], boundaries: [Float], destination: vImage.PixelBuffer<Format>)`

Applies a set of piecewise polynomials to a 2-channel, 32-bit interleaved buffer.

```
func applyPolynomial(coefficientSegments: [[Float]], boundaries: [Float
], destination: vImage.PixelBuffer<Format>)
```

Applies a set of piecewise polynomials to a 4-channel, 32-bit interleaved buffer.