

Transforming with a gamma function

Use gamma functions to apply color transformations to images.

Overview


Gamma correction functions correct the brightness profile of an image by applying an exponent to each pixel in an image. The piecewise gamma correction functions allow you to add a linear transformation to pixels below a certain boundary. You can use gamma correction functions to prepare an image to display or print on a particular device.

Topics

Applying a gamma function

```
func vImageCreateGammaFunction(Float, Int32, vImage_Flags) -> GammaFunction!
```

Returns a gamma function object.

 Gamma function types

Types of full- or half-precision gamma functions.

```
func vImageGamma_Planar8toPlanarF(UnsafePointer<vImage_Buffer>, UnsafePointer<vImage_Buffer>, GammaFunction!, vImage_Flags) -> vImage_Error
```

Applies a gamma function to an 8-bit planar image to produce a 32-bit planar image.

```
func vImageGamma_PlanarFtoPlanar8(UnsafePointer<vImage_Buffer>, UnsafePointer<vImage_Buffer>, GammaFunction!, vImage_Flags) -> vImage_Error
```

Applies a gamma function to a 32-bit planar image to produce an 8-bit planar image.

```
func vImageGamma_PlanarF(UnsafePointer<vImage_Buffer>, UnsafePointer<vImage_Buffer>, GammaFunction!, vImage_Flags) -> vImage_Error
```

Applies a gamma function to a PlanarF image.

```
func vImageDestroyGammaFunction(GammaFunction!)
```

Destroys a gamma function object.

{ } Adjusting saturation and applying tone mapping

Convert an RGB image to discrete luminance and chrominance channels, and apply color and contrast treatments.

Applying a piecewise gamma function

{ } Adjusting the brightness and contrast of an image

Use a gamma function to apply a linear or exponential curve.

```
func vImagePiecewiseGamma_Planar8(UnsafePointer<vImage_Buffer>, UnsafePointer<vImage_Buffer>, UnsafePointer<Float>, Float, UnsafePointer<Float>, Pixel_8, vImage_Flags) -> vImage_Error
```

Applies a piecewise gamma function to transform an 8-bit planar image.

```
func vImagePiecewiseGamma_Planar8toPlanar16Q12(UnsafePointer<vImage_Buffer>, UnsafePointer<vImage_Buffer>, UnsafePointer<Float>, Float, UnsafePointer<Float>, Pixel_8, vImage_Flags) -> vImage_Error
```

Applies a piecewise gamma function to transform an 8-bit planar image to a 16Q12 planar image.

```
func vImagePiecewiseGamma_Planar8toPlanarF(UnsafePointer<vImage_Buffer>, UnsafePointer<vImage_Buffer>, UnsafePointer<Float>, Float, UnsafePointer<Float>, Pixel_8, vImage_Flags) -> vImage_Error
```

Applies a piecewise gamma function to transform an 8-bit planar image to a 32-bit planar image.

```
func vImagePiecewiseGamma_Planar16Q12(UnsafePointer<vImage_Buffer>, UnsafePointer<vImage_Buffer>, UnsafePointer<Float>, Float, UnsafePointer<Float>, Pixel_16S, vImage_Flags) -> vImage_Error
```

Applies a piecewise gamma function to transform a 16Q12 planar image.

```
func vImagePiecewiseGamma_Planar16Q12toPlanar8(UnsafePointer<vImage_Buffer>, UnsafePointer<vImage_Buffer>, UnsafePointer<Float>, Float, UnsafePointer<Float>, Pixel_16S, vImage_Flags) -> vImage_Error
```

Applies a piecewise gamma function to transform a 16Q12 planar image to an 8-bit planar image.

```
func vImagePiecewiseGamma_PlanarF(UnsafePointer<vImage_Buffer>, UnsafePointer<vImage_Buffer>, UnsafePointer<Float>, Float, UnsafePointer<Float>, Float, vImage_Flags) -> vImage_Error
```

Applies a piecewise gamma function to transform a 32-bit planar image.

```
func vImagePiecewiseGamma_PlanarFtoPlanar8(UnsafePointer<vImage_Buffer>, UnsafePointer<vImage_Buffer>, UnsafePointer<Float>, Float, UnsafePointer<Float>, Float, vImage_Flags) -> vImage_Error
```

Applies a piecewise gamma function to transform a 32-bit planar image to an 8-bit planar image.

Applying a symmetric piecewise gamma function

```
func vImageSymmetricPiecewiseGamma_Planar16Q12(UnsafePointer<vImage_Buffer>, UnsafePointer<vImage_Buffer>, UnsafePointer<Float>, Float, UnsafePointer<Float>, Pixel_16S, vImage_Flags) -> vImage_Error
```

Applies a symmetric piecewise gamma function to transform a 16Q12 planar image to an 8-bit planar image.

```
func vImageSymmetricPiecewiseGamma_PlanarF(UnsafePointer<vImage_Buffer>, UnsafePointer<vImage_Buffer>, UnsafePointer<Float>, Float, UnsafePointer<Float>, Float, vImage_Flags) -> vImage_Error
```

Applies a symmetric piecewise gamma function to transform a 32-bit planar image.

See Also

Applying color transforms to images

☰ Transforming with lookup tables

Use lookup tables to apply color transformations to images.

☰ Transforming with polynomials

Use polynomials to apply color transformations to images.

☰ Transforming with matrix multiplication

Use matrix multiplication to apply color transformations to images.

☰ Applying a flood fill to an image

Fill connected components of an image with a new color.