

Calculations

CalcColorSpaceTransform

- Funktionen zum Umrechnen von Farbmodellen.

Für einzelne Tripel:

```
export function cvtRGBtoXYZ(rgb_array, colorSpace =  
"709") {...}  
export function cvtHSVtoRGB(HSV_array) {...}  
export function cvtRGBtoHSV(RGB_array) {...}  
  
export function cvtXYZtoxy(XYZ_array) {...}  
export function cvtXYZtoxyY(XYZ_array) {...}
```

Für Singal-Arrays:

```
export function cvtSignalRGBtoXYZ(signalRGB, colorSpace =  
"709") {...}  
  
export function cvtSignalXYZtoxy(signalXYZ) {...}  
export function cvtSignalXYZtoxyY(signalXYZ) {...}
```

CalcComponentSignal

- Funktionen zum Rechnen mit Komponentensignalen.

Für einzelne Tripel:

```
export function cvtRGBtoYCBCR(RGB, standard = "709") {...}  
export function cvtYCBCRtoRGB(YCBCR, standard = "709") {...}  
  
export function upscaleYCBCR(YCBCR, bitDepth = 10) {...}  
export function downscaleYCBCR(YCBCR, bitDepth = 10) {...}
```

```
export function limiterYCBCR(YCBCR, bitDepth, fullVideoData = false) {...}
```

Für Singal-Arrays:

```
export function cvtSignalRGBtoYCBCR(signalRGB, videoStandard = "709" ) {...}  
export function cvtSignalYCBCRtoRGB(signalYCBCR, videoStandard = "709" ) {...}  
  
export function upscaleSignalYCBCR(signalSmallYCBCR, bitDepth = 10 ) {...}  
export function downscaleSignalYCBCR(signalYCBCR, bitDepth = 10 ) {...}  
  
export function limiterSignalYCBCR(signalYCBCR, bitDepth, fullVideoData = false) {...}  
export function limiterSignalSmallRGB(signalSmallRGB, fullVideoData = false) {...}
```

CalcHelpers

- Sonstige Hilfs-Funktionen.

```
export function rgbToString(rgbArray){...}  
export function rgbToComplColorString(rgbArray){...}  
export function clamp(value, min = 0, max = 1) {...}
```

CalcRGBSignal

- Funktionen des Correctors.

Für einzelne Tripel:

```
export function upscaleRGB(RGB, bitDepth = 10) {...}  
export function downscaleRGB(RGB, bitDepth = 10) {...}
```

Für Singal-Arrays:

```
export function upscaleSignalRGB(signalSmallRGB, bitDepth = 10  
) {...}  
export function downscaleSignalRGB(signalRGB, bitDepth = 10 )  
{...}
```

CalcSignalCorrector

- Funktionen des Correctors.

Für einzelne Tripel:

```
function offsetContrast(pixelValue = [0, 0, 0], m = 1) {...}  
function offsetBrightness(pixelValue = [0, 0, 0], b = 0) {...}  
function offsetGamma(pixelValue = [0, 0, 0], gamma = 1,  
maxValue = 1) {...}
```

Für Singal-Arrays:

```
export function offsetSignalContrast(signalRGB, m = 1) {...}  
export function offsetSignalBrightness(signalRGB, b = 0) {...}  
export function offsetSignalGamma(signalRGB, gamma = 1,  
maxValue = 1) {...}
```

CalcSignalGenerator

- Funktionen der Signal-Generatoren.

```
export function generateRGBSignalFullColor(valueRGB, width, height){...}  
export function generateRGBSignalBars(width = 8, height = 1, type100 = true){...}  
export function generateRGBSignalGradient(startRGB, endRGB, width, height, directionHorizontal=true){...}  
  
function blendColor(firstRGB, secondRGB, ratio = 0.5){...}
```

CalcHelpers

- Sonstige Hilfs-Funktionen.

```
export function rgbToString(rgbArray){...}  
  
export function rgbToComplColorString(rgbArray){...}  
  
export function clamp(value, min = 0, max = 1) {...}
```
