

# IceMan - Data Conversion

## **ice.Image Dither**(*amplitude*, *seed*)

Dither an image suitable to quantizing to a lower bit depth.

### Parameters

*amplitude*

A random number "uniformly" distributed in the range [-amplitude, amplitude] is added to the image (float).

*seed*

Integer number to seed the random number generator (int).



There is no actual type-conversion. If you need type conversion, you must use additionally call **TypeConvert**.

### Example

```
t = floatImage.Dither(1.0/256.0, 1) display = t.TypeConvert(ice.constants.FRACTIONAL)
```

## **ice.Image D1ToRGB**(*colorStandard*)

The image is assumed to be legal D1 (720x486, two channel YUV/YIQ, chroma-subsampled), and the result is an RGB image, appropriately gamma corrected.

### Parameters

*colorStandard*

One of *IceColorStandard* (int):

- ice.constants.PAL
- ice.constants.NTSC



Gamma correction values are different for PAL and NTSC, as are some intermediate color spaces.

## **ice.Image RGBToD1**(*colorStandard*)

Return an image encoded in D1 (720x486, two-channel YUV/YIQ, chroma-subsampled, gamma corrected) format.

### Parameters

*colorStandard*

PAL or NTSC (int).



This operation performs more than a simple data conversion!

## **ice.Image TypeConvert(*destinationType*)**

Convert an image to a different type. Attempts to convert to the same type are legal, but do nothing.

### **Parameters**

*destinationType*

Data type to convert to one of IceComponents constants:

- ice.constants.FRACTIONAL
- ice.constants.FIXED\_POINT
- ice.constants.FLOAT
- ice.constants.DOUBLE