Color Space Conversions

This section includes functions for performing conversions between different color spaces.

[res] image.rgb2lab([dst,] src)

Converts a `src` RGB image to Lab.
If `dst` is provided, it is used to store the output image. Otherwise, returns a new `res` Tensor.

[res] image.lab2rgb([dst,] src)

Converts a `src` Lab image to RGB.
If `dst` is provided, it is used to store the output image. Otherwise, returns a new `res` Tensor.

[res] image.rgb2yuv([dst,] src)

Converts a RGB image to YUV. If `dst` is provided, it is used to store the output image. Otherwise, returns a new `res` Tensor.

[res] image.yuv2rgb([dst,] src)

Converts a YUV image to RGB. If `dst` is provided, it is used to store the output image. Otherwise, returns a new `res` Tensor.

[res] image.rgb2y([dst,] src)

Converts a RGB image to Y (discard U and V).
If `dst` is provided, it is used to store the output

image. Otherwise, returns a new `res` Tensor.

[res] image.rgb2hsl([dst,] src)

Converts a RGB image to HSL.
If `dst` is provided, it is used to store the output
image. Otherwise, returns a new `res` Tensor.

[res] image.hsl2rgb([dst,] src)

Converts a HSL image to RGB.
If `dst` is provided, it is used to store the output
image. Otherwise, returns a new `res` Tensor.

[res] image.rgb2hsv([dst,] src)

Converts a RGB image to HSV.
If `dst` is provided, it is used to store the output
image. Otherwise, returns a new `res` Tensor.

[res] image.hsv2rgb([dst,] src)

Converts a HSV image to RGB.
If `dst` is provided, it is used to store the output
image. Otherwise, returns a new `res` Tensor.

[res] image.rgb2nrgb([dst,] src)

Converts an RGB image to normalized-RGB.

[res] image.y2jet([dst,] src)

Converts a L-levels (1 to L) greyscale image into a L-levels jet heat-map.
If `dst` is provided, it is used to store the output image. Otherwise, returns a new `res` Tensor.

This is particulary helpful for understanding the magnitude of the values of a matrix, or easily spot peaks in scalar field (like probability densities over a 2D area).
For example, you can run it as

```
image.display{image=image.y2jet(torch.linspace(1,10,10)), zoom=50}
```