“Read me to use better color space”

we reviewed three very common color spaces in computer vision: RGB, HSV, and L*a*b*.

The RGB color space is the most common color space in computer vision. It’s an additive color space, where colors are defined based on combining values of red, green, and blue.

While quite simple, the RGB color space is unfortunately unintuitive for defining colors as it’s hard to pinpoint *exactly* how much red, green, and blue compose a certain color — imagine looking a specific region of a photo and trying to identify how much red, green, and blue there is using only your naked eye!

Luckily, we have the HSV color space to compensate for this problem. The HSV color space is also intuitive, as it allows us to define colors along a *cylinder* rather than a RGB cube. The HSV color space also gives lightness/whiteness its own separate dimension, making it easier to define shades of color.

However, both the RGB and HSV color spaces fail to mimic the way humans perceive color — *there is no way to mathematically define how perceptually different two arbitrary colors are using the RGB and HSV models*.

And that’s exactly why the L*a*b* color space was developed. While more complicated, the L*a*b* provides with perceptual uniformity, meaning that the distance between two arbitrary colors has actual meaning.

All that said, you’ll find that you will use the RGB color space for most computer vision applications. While it has many shortcomings, you cannot beat its simplicity — it’s simply adequate enough for most systems.

There will also be times when you use the HSV color space — particularly if you are interested in tracking an object in an image based on its *color*. It’s very easy to define color ranges using HSV.

For basic image processing and computer vision you likely won’t be using the L*a*b* color space that often. But when you’re concerned with color management, color transfer, or color consistency across multiple devices, the L*a*b* color space will be your best friend. It also makes for an excellent color image descriptor.

Finally, we discussed converting an image from RGB to grayscale. While the grayscale representation of an image is not technically a color space, it’s worth mentioning in the same vein as RGB, HSV, and L*a*b*. We often use grayscale representations of an image when color is not important — this allows us to conserve memory and be more computationally efficient.

I give you all colors conversation syntax in “colorSpaces.py” code comments

Mail me your issue : hosseinsamadi22@gmail.com

کاربرد هر فضای رنگ:

(96463128 _ حسین صمدی _تمرین اول بینایی ماشین)

قبل از عملیات segmentation و imageProcess PipLine با فضاهای رنگ روبه‌رو می‌شویم :colorspaces

کتابخانه openCV فضاهای رنگ ← 1.RGB 2.HSV 3.L*a*b 4.grayscale

ما برای استخراج دیتاهای مورد نیاز می‌توانیم از فضاها یا فیلترهای رنگ بالا استفاده کنیم.

RGB = (قرمز ، سبز ، آبی) = ([0:255] ، [0:255] ، [0:255]) «مکعبی»

که می‌توان با بهره‌برداری از numpy مقدار شدت هر رنگ را دریافت و تغییر داد.

HSV = (Hue ، saturation ، Value) = (رنگ خلوص ، رنگی شدت ، روشنی حجم) «فضای رنگ استوانه‌ای»

L*a*b = (L-channel ، a-channel ، b-channel) = (pixel روشنایی ، سبز-قرمز میزان ، آبی-زرد میزان) «کره‌ای»

gray scale = همان فضای RGB را به صورت خاکستری نمایش می‌دهد.

زمانی که می‌خواهیم محاسباتی انجام دهیم به فضای gray می‌رویم که رنگ در آن مهم نیست.

RGB رایج‌ترین فضای رنگ در بینایی ماشین می‌باشد در این محیط می‌توان میزان رنگ‌ها را دقیق مشخص کرد

HSV روشنایی/سفیدی جداگانه به خود می‌دهد که می‌توان در آن سایه‌های رنگ را مشخص کرد.

L*a*b یکنواختی ادراکی را ارائه می‌دهد که فاصله بین دو رنگ معنی واقعی دارد.

Sahand