

SSIM (Structure Similarity Index Measure) 结构衡量指标+代码

好风

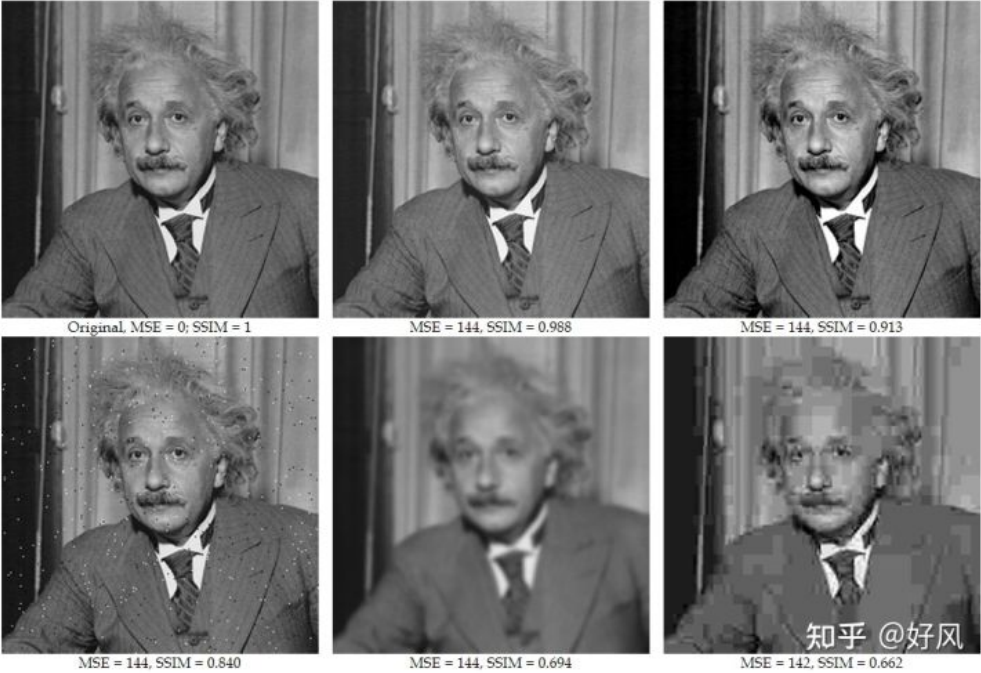
22 人赞同了该文章

SSIM (Structure Similarity Index Measure) 结构衡量指标+代码

介绍

结构相似指标可以衡量图片的失真程度，也可以衡量两张图片的相似程度。与MSE和PSNR衡量绝对误差不同，SSIM是感知模型，即更符合人眼的直观感受。

同样MSE下，不同SSIM展现的图片结果：



SSIM的取值范围[-1, 1], 具有对成性，边界性，唯一最大性（当且仅当x=y时SSIM=1），是一种距离公式。

SSIM理论

SSIM 主要考量图片的三个关键特征：亮度（Luminance）, 对比度（Contrast）, 结构（Structure）

Luminance

亮度以平均灰度衡量，通过平均所有像素的值得到。

$$\mu_x = \frac{1}{N} \sum_{i=1}^N x_i$$

对比函数

$$l(x, y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}$$

Contrast

对比度通过灰度标准差来衡量。标准差无偏估计：

$$\sigma_x = \left(\frac{1}{N-1} \sum_{i=1}^N (x_i - \mu_x)^2 \right)^{\frac{1}{2}}$$

$$l(x,y)=\frac{2\sigma_x\sigma_y+C_2}{\sigma_x^2+\sigma_y^2+C_2}$$

Structure

结构对比比较的是经过归一化后 $x-\mu_x/\sigma_x$ 与 $(y-\mu_y)/\sigma_y$ 的比较,即可以用相关性系数衡量

$$c(x,y)=\frac{2\sigma_{xy}+C_3}{\sigma_x\sigma_y+C_3}$$

$$\sigma_{xy}=\frac{1}{N-1}\sum_{i=1}^N(x_i-\mu_x)(y_i-\mu_y)$$

C_1 常量避免 $l(x,y)$ 的值接近0时不稳定。 C_2,C_3 同。

$C_1=(K_1L)^2$, $C_2=(K_2L)^2$, $C_3=C_2/2$, 经验常取 $K_1=0.01$, $K_2=0.03$, L 像素动态取值范围 $L=2^{bits\ per\ pixel}-1$

SSIM

$$S(x,y)=l(x,y)^{\alpha}\cdot c(x,y)^{\beta}\cdot s(x,y)^{\gamma}$$

α,β,γ 分别代表了不同特征在SSIM衡量中的占比,当都为1时,有：

$$SSIM(x,y)=\frac{(2\mu_x\mu_y+C_1)(2\sigma_{xy}+C_2)}{(\mu_x^2+\mu_y^2+C_1)(\sigma_x^2+\sigma_y^2+C_2)}$$

SSIM的实践常用方法MSSIM (Mean SSIM)

实际上,当需要衡量一整张图片的质量,经常使用的是以一个一个窗口计算SSIM然后求平均。

当我们用一个一个block去计算平均值,标准差,协方差时,这种方法容易造成 **blocking artifacts**,所以在计算MSSIM时,会使用到 **circular-symmetric Gaussian weighting function** 圆对称的高斯加权公式 $w=w_i|i=1,2,...,N$,标准差为1.5,和为1,来估计局部平均值,标准差,协方差。

代码：

```
def gaussian(window_size, sigma):
    gauss = torch.Tensor([exp(-(x - window_size//2)**2/float(2*sigma**2)) for x in range(window_size)])
    return gauss/gauss.sum()

def create_window(window_size, channel):
    _1D_window = gaussian(window_size, 1.5).unsqueeze(1)
    _2D_window = _1D_window.mm(_1D_window.t()).float().unsqueeze(0).unsqueeze(0)
    window = Variable(_2D_window.expand(channel, 1, window_size, window_size).clone())
    return window
```

SSIM:

$$SSIM(x,y)=\frac{(2\mu_x\mu_y+C_1)(2\sigma_{xy}+C_2)}{(\mu_x^2+\mu_y^2+C_1)(\sigma_x^2+\sigma_y^2+C_2)}$$

需要计算：

$$\mu_x=\sum_{i=1}^Nw_ix_i\sigma_x=(\sum_{i=1}^Nw_i(x_i-\mu_x)^2)^{\frac{1}{2}}\sigma_{xy}=\sum_{i=1}^Nw_i(x_i-\mu_x)(y_i-\mu_y)$$

```
def _ssim(img1, img2, window, window_size, channel, size_average = True):
    mu1 = F.conv2d(img1, window, padding = window_size//2, groups = channel)
    mu2 = F.conv2d(img2, window, padding = window_size//2, groups = channel)

    mu1_sq = mu1.pow(2)
    mu2_sq = mu2.pow(2)
    mu1_mu2 = mu1*mu2

    sigma1_sq = F.conv2d(img1*img1, window, padding = window_size//2, groups = channel)
    sigma2_sq = F.conv2d(img2*img2, window, padding = window_size//2, groups = channel)
    sigma12 = F.conv2d(img1*img2, window, padding = window_size//2, groups = channel)

    C1 = 0.01**2
    C2 = 0.03**2

    ssim_map = ((2*mu1_mu2 + C1)*(2*sigma12 + C2))/((mu1_sq + mu2_sq + C1)*(sigma1_sq + sigma2_sq + C1))

    if size_average:
        return ssim_map.mean()
    else:
        return ssim_map.mean(1).mean(1).mean(1)
```

经验常取窗口大小为11x11.

$$MSSIM(X,Y)=\frac{1}{M}\sum_{j=1}^MSSIM(x_i,y_j)$$



$$\sigma^2 = \frac{\sum (X - \mu)^2}{N} \tag{1}$$
$$= \frac{\sum (X^2 - 2\mu X + \mu^2)}{N} \tag{2}$$
$$= \frac{\sum X^2}{N} - \frac{2\mu \sum X}{N} + \frac{N\mu^2}{N} \tag{3}$$
$$= \frac{\sum X^2}{N} - 2\mu^2 + \mu^2 \tag{4}$$
$$= \frac{\sum X^2}{N} - \mu^2 \tag{5}$$

知乎 @好风

同理

$$\sigma_{xy} = \frac{1}{N} \sum_i x_i y_i - \mu_x \mu_y$$

个人理解：

不同于MSE, SSIM以窗口的局部特征，在加上高斯核，相当于引入了平滑先验，符合人对视觉的感知。在用MSE作为loss的时候，可以额外加入SSIM，效果会更好。

difference SSIM, DSSIM = (1 - SSIM)/2， 范围[0, 1]

参考

https://www.cns.nyu.edu/pub/eero/wang03-reprint.pdf
www.cns.nyu.edu/pub/eero/wang03-reprint.pdf

https://medium.com/srm-mic/all-about-structural-similarity-index-ssim-theory-code-in-pytorch-...
medium.com/srm-mic/all-about-structural-similarity-in...

文武：图片结构相似性算法：SSIM
46 赞同 · 3 评论 文章

https://ece.uwaterloo.ca/~z70wang/research/ssim/
ece.uwaterloo.ca/~z70wang/research/ssim/

Po-Hsun-Su/pytorch-ssim: pytorch structural similarity (SSIM) loss (github.com)
github.com

发布于 2021-08-12 22:53

算法 计算机视觉

7 条评论

切换为时间排序

写下你的评论...

LauChenXin

2021-10-22

S公式错了吧

赞

好风 (作者) 回复 LauChenXin

2021-11-06

感谢纠正，但可否在具体一点？

赞

---

2021-08-12



赞

知乎用户

06-15

输入图像是0-255 还是需要归一化呢

赞

不停吓

04-29

高斯加权公式加入时，是经验情况下一般为1.5和1吗

赞

TingKo.1

03-08



均值非负数，方差非负数，C也为非负数，所以SSIM范围应该是前者

赞

推荐阅读

图像质量评价之结构相似性SSIM（上）

本文总结归纳自这篇论文：image quality assessment: from error visibility to structural similarity概述这篇文章主要介绍对 图像质量进行打分评价的一个很经典的指数——结构相似性（str...

gdymi... 发表于译智社



如何度量两幅图像的相似度--结构相似度SSIM 原理及代码

Tina姐 发表于医学图像人...



图片的相似度--图像结构相似度SSIM

吃菜的菜狗

图像质量评价（四）：SSIM和MSSIM

SSIMSSIM是一个广泛使用的图像质量评价指标，它是基于人眼观看图像时会提取其中的结构化信息的假设。SSIM是一种全参考的评价方法，对于图像x和图像y，其SSIM计算方式如下: 上式中l(x,y)是...

寒衣 发表于图像质量评...

