

## 1、直方图方法

方法描述：有两幅图像patch(当然也可是整幅图像)，分别计算两幅图像的直方图，并将直方图进行归一化，然后按照某种距离度量的标准进行相似度的测量。

方法的思想：基于简单的向量相似度来对图像相似度进行度量。

优点：直方图能够很好的归一化，比如256个bin条，那么即使是不同分辨率的图像都可以直接通过其直方图来计算相似度，计算量适中。比较适合描述难以自动分割的图像。

缺点：直方图反应的是图像灰度值得概率分布，并没有图像的空间位置信息在里面，因此，常常出现误判；从信息论来讲，通过直方图转换，信息丢失量较大，因此单一的通过直方图进行匹配显得有点力不从心。

## 2、图像模板匹配

一般而言，源图像与模板图像patch尺寸一样的话，可以直接使用上面介绍的图像相似度测量的方法；如果源图像与模板图像尺寸不一样，通常需要进行滑动匹配窗口，扫面个整幅图像获得最好的匹配patch。

在OpenCV中对应的函数为：`matchTemplate()`：函数功能是在输入图像中滑动窗口寻找各个位置与模板图像patch的相似度。

## 3、PSNR峰值信噪比

PSNR (Peak Signal to Noise Ratio)，一种全参考的图像质量评价指标。

简介：[https://en.wikipedia.org/wiki/Peak\\_signal-to-noise\\_ratio](https://en.wikipedia.org/wiki/Peak_signal-to-noise_ratio)

PSNR是最普遍和使用最为广泛的一种图像客观评价指标，然而它是基于对应像素点间的误差，即基于误差敏感的图像质量评价。由于并未考虑到人眼的视觉特性（人眼对空间频率较低的对比差异敏感度较高，人眼对亮度对比差异的敏感度较色度高，人眼对一个区域的感知结果会受到其周围邻近区域的影响等），因而经常出现评价结果与人的主观感觉不一致的情况。

## 4、SSIM (structural similarity) 结构相似性

也是一种全参考的图像质量评价指标，它分别从亮度、对比度、结构三方面度量图像相似性。

SSIM取值范围 $[0, 1]$ ，值越大，表示图像失真越小。

在实际应用中，可以利用滑动窗将图像分块，令分块总数为 $N$ ，考虑到窗口形状对分块的影响，采用高斯加权计算每一窗口的均值、方差以及协方差，然后计算对应块的结构相似度SSIM，最后将平均值作为两图像的结构相似性度量，即平均结构相似性MSSIM：

## 5、感知哈希算法

(perceptual hash algorithm)

<http://blog.csdn.net/fengbingchun/article/details/42153261>

感知哈希~~算法~~(perceptual hash algorithm)，它的作用是对每张图像生成一个“指纹”(fingerprint)字符串，然后比较不同图像的指纹。结果越接近，就说明图像越相似。

实现步骤：

1. 缩小尺寸：将图像缩小到 $8*8$ 的尺寸，总共64个像素。这一步的作用是去除图像的细节，只保留结构/明暗等基本信息，摒弃不同尺寸/比例带来的图像差异；
2. 简化色彩：将缩小后的图像，转为64级灰度，即所有像素点总共只有64种颜色；
3. 计算平均值：计算所有64个像素的灰度平均值；
4. 比较像素的灰度：将每个像素的灰度，与平均值进行比较，大于或等于平均值记为1，小于平均值记为0；
5. 计算哈希值：将上一步的比较结果，组合在一起，就构成了一个64位的整数，这就是这张图像的指纹。组合的次序并不重要，只要保证所有图像都采用同样次序就行了；
6. 得到指纹以后，就可以对比不同的图像，看看64位中有多少位是不一样的。在理论上，这等同于“汉明距离”(Hamming distance,在信息论中，两个等长字符串之间的汉明距离是两个字符串对应位置的不同字符的个数)。如果不相同的数据位数不超过5，就说明两张图像很相似；如果大于10，就说明这是两张不同的图像。

以上内容摘自：

<http://www.ruanyifeng.com/blog/2011/07/principle of similar image search.html>