Understanding ISP Pipeline - Downsample

刘斯宁
Camera技术专家

已关注

51 人赞同了该文章

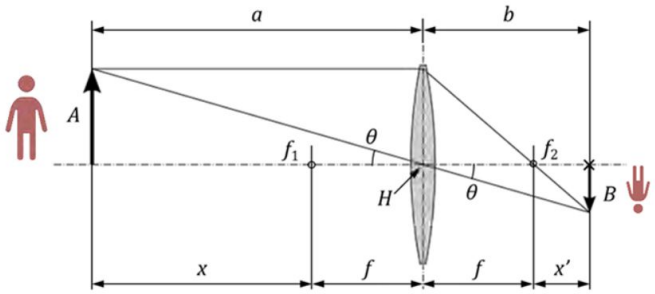
应热心网友的邀请，特别写一期关于downsample的文章，希望能够对新老用户都有一定的启发。

对于一个相机来说，镜头的主要作用是收集物方空间（object space）的光线，然后按照一定的函数关系把光线投射到像方空间（image space）。显然，物方空间的会存在一些人们感兴趣的物体（object），而像方空间的焦平面上，会有一个image sensor，它负责把光信号转换成电信号，再经过AD转换后变成一个个数字记录下来。

光学系统的放大率

相机上的镜头是一个光学系统。根据几何光学的基本原理，光学系统在成像的过程中，存在三种放大作用。

1. 横向放大率 (lateral magnification)



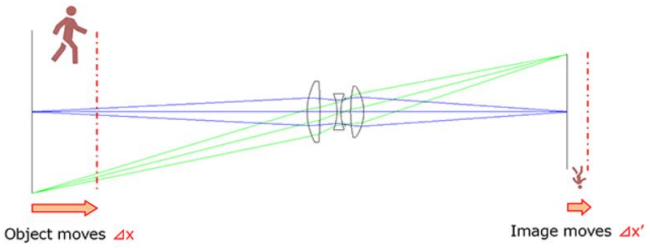
横向放大率

横向放大率一般用符号 β 表示，定义为像高 B 与物高 A 之比.

$$\beta = \frac{B}{A}$$

横向放大率公式

2. 轴向放大率 （longitudinal magnification）



轴向放大率示意图

当一个物体沿光轴移动一个小距离Δx 时，它的像也会沿光轴移动一个小距离Δx'。另一种等价的说法是，一个宽度为Δx 的物体，经过光学系统成像后，像的宽度为Δx'。

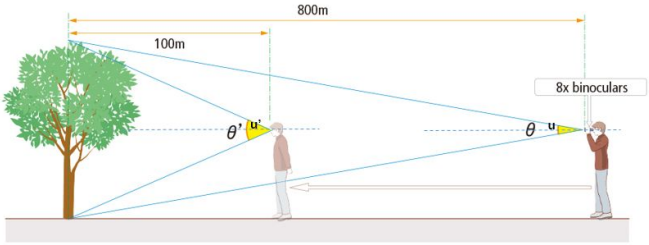
轴向放大率一般用符号α 表示，定义为像宽Δx' 与物宽Δx 之比。

$$\alpha = \frac{\Delta x'}{\Delta x}$$

轴向放大率公式

3. 角放大率 （angular magnification）

对于显微镜、望远镜等产品来说，光学系统需要放大的是入射光线相对于光轴的张角，相当于把人眼移到离物体更近的地方去观察。



角放大率示意图

角放大率一般用γ表示，定义为出射光线和光轴的夹角u' 与入射光线和光轴的夹角u 的比值。

$$\gamma = \frac{u'}{u} = \frac{n}{n'}\frac{1}{\beta}$$

角放大率公式

镜头的放大倍数

每个镜头，在任一时刻，都会有一个确定的放大倍数，其意义就是横向放大率，对于显微和望远镜头则是角放大率。

举例来说，某CMOS sensor 像素阵列的实际宽度是2.4cm，使用某镜头时，可以拍摄到的最大视场宽度是7.1cm，则该镜头（此时此刻）的放大倍率是2.4/7.1=0.34倍，用比数方式可以表示为1:3.0。显然，这是一个放大率小于1的镜头，视场宽度大于CMOS像素阵列的宽度。

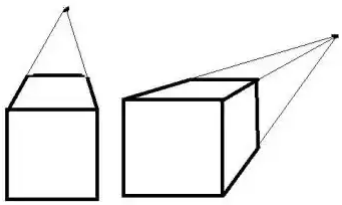
镜头的放大倍率原理

拍摄小昆虫所需的微距镜头则需要放大率大于1。

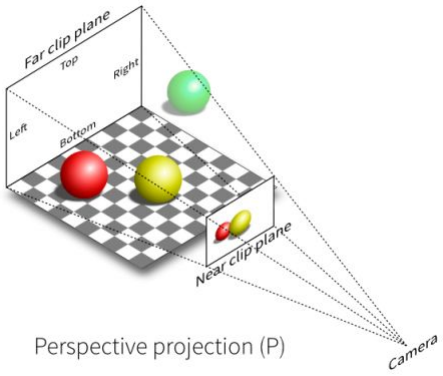
镜头的放大倍率

镜头的放大倍数会有一个厂家标称的规格，指的是这个镜头在指定的焦段内，在成清晰像的前提下，能够获得的最大放大倍数。

显然，对于普通镜头来说，成像的放大率与物体离镜头的远近有关，距离越远放大率越小。当物体本身存在一定厚度时，比如人的鼻子相对于脸部会突出一定的距离，这就导致同一物体的不同部位得到不一致的放大倍数，产生的效果叫做透视畸变。



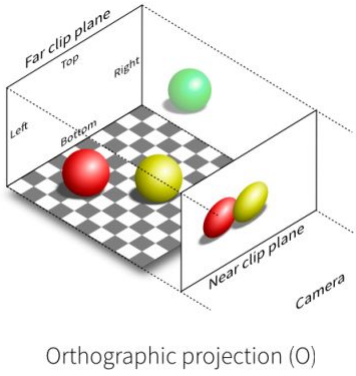
透视畸变原理



普通镜头的成像原理

然而也有例外，机器视觉行业会用到一种远心镜头，它采用的是望远镜的成像原理，其特点是在一定范围内，成像的放大率与物体离镜头的远近无关，这样通过图像测量到的数据能够保持稳定，不需要考虑物体距离镜头的远近导致的缩放，也不需要考虑物体本身的高度导致的缩放问题。

远心镜头



远心镜头的成像原理

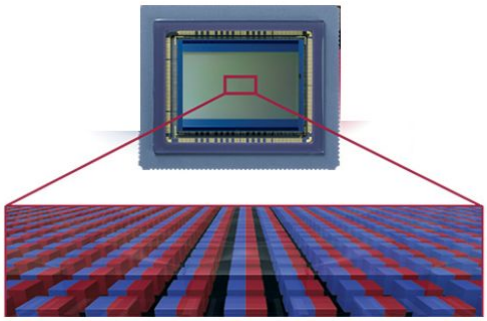
采样 Sampling

连续信号在时间（或空间）上以某种方式变化着，而采样的过程就是在时间（或空间）上，以T为单位间隔来测量连续信号的值。T称为采样间隔，而T的倒数称为采样率。

对于时间（temporal）上的采样，根据需求的不同，T 经常以毫秒ms、微秒μs为单位，而对应的采样率则是以kHz、MHz为单位。

对于空间（spatial）上的采样，根据需求的不同，T 经常以毫米mm、微米μm为单位，而对应的采样率则是用samples/meter、samples/mm为单位。在印刷行业会经常使用DPI 单位（dots per inch）。

手机和安防监控上使用CMOS sensor 像素大小通常是1~2μm左右。以2μm 为例，如果sensor 阵列的宽度是2048 个像素，则成像阵列的实际尺寸是4mm。



当镜头的放大倍数是0.3时，每个像素（2μmx2μm）会映射到物方空间6.66μm x 6.66μm 的面积，换句话说，从这个物方单位面积上发出的光信号，进入相机后，都会被这一个像素所采集。

降采样 Downsample

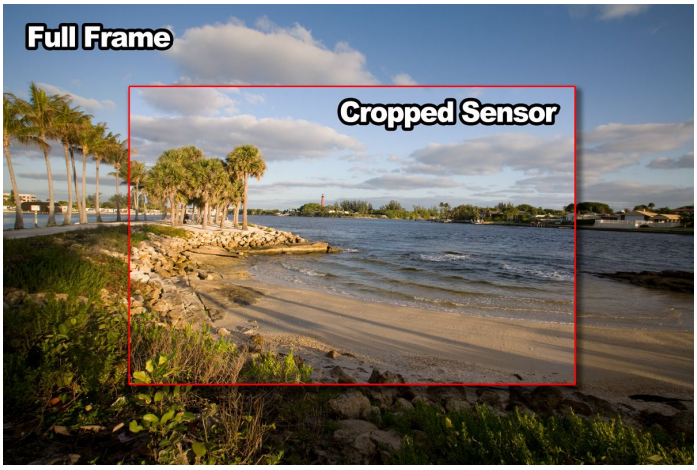
人们对图像质量的追求是无止境的，因此各种图像处理系统对图像信号进行采样的频率，包括时间频率（帧率）和空间频率（分辨率）都呈现出越来越高的趋势。

然而从业务需求的角度看，并不是所有的图像处理任务都必须使用最高帧率、最高分辨率的图像。对于同一个拍摄场景，有些任务每隔1秒处理一次就可以，有些任务使用很低的分辨率（如CIF）就可以很好地完成。因此，相机产品中经常需要对CMOS sensor 输出的高帧率、高分辨率原始图像进行降采样，以获得多种低帧率、低分辨率的图像流，来适配不同处理任务的实际需求，在满足业务需求的基础上实现节省成本（传输带宽、存储空间、CPU算力、芯片面积、功耗等）。

实际上，CMOS sensor 本身就会支持一些降采样的方法，最常见的是cropping、decimation、binning 三种方式。

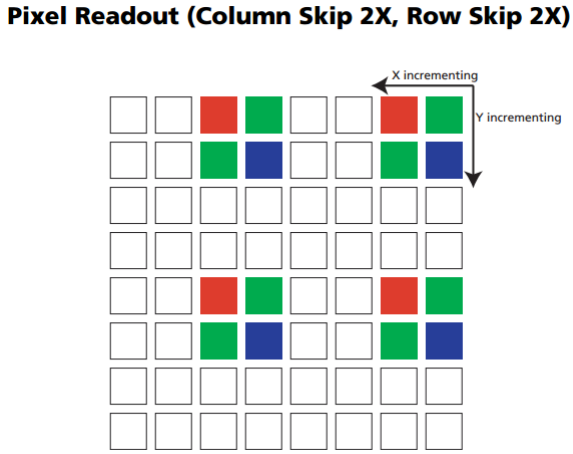
cropping 的本质是把一个高分辨率的sensor当成一个低分辨率的sensor用，比如通过sensor参数配置，让一个500W像素的sensor只输出200W像素。由于sensor输出像素的速率是一定的，当每帧的分辨率变小时，帧率就可以变高。

如果单纯从图像质量的角度考虑，让5MP sensor 只输出2MP 其实并不是一个特别有价值的做法，因为不同的sensor 像素大小和信噪比特性是不同的，如果sensor感光阵列的面积和像素工艺都相同，则2MP型号像素可以更大，从而信噪比更好，成像质量也就更好。另外，sensor crop之后视场角会变小，这就会对选配镜头的工作带来一定的困扰。但是从商业角度考虑，这种做法有时又有一定的意义，比如在2MP 型号sensor 缺货或者价格不合适的时候，不妨用5MP的产品线临时顶替一下，实际上很多厂家都这么做过。



sensor cropping

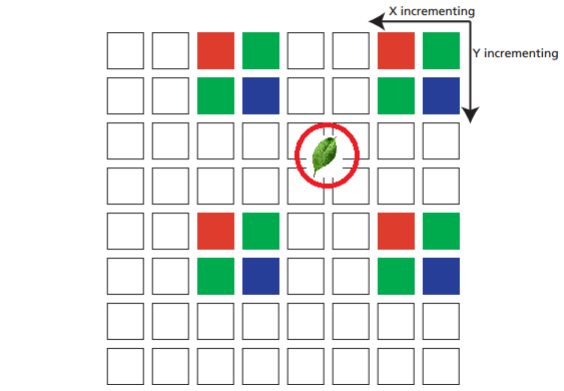
decimation 在很多资料中也叫做skipping，指的是每采样一个点就跳过若干个点，使采样率按照一定的整数比例下降。



sensor decimation/skipping

decimation 的最初来源是古罗马军队的一个军令，如果某个部队发生了叛变、逃离战场、丢失军旗等问题，就要接收严厉的惩罚，不管是将领还是士兵，不管是带头逃跑的懦夫，还是死守战场的荣耀战士，全部人都要参加抽签，每十个人中抽出一个杀死，以儆效尤。

decimation 方式的特点是不改变视场角大小，由于每帧的像素数量变少，所以sensor输出的帧率可以提高。这两个特征都是有利的。但是也存在不利的一面，就是被跳过的像素所包含的信息会完全丢失，不会在输出图像中留下任何痕迹。当人们预期画面中应该出现一个线条，但这条线却没有出现时，人们会下意识地感觉这个画面很不正常，这会影响图像主观质量的评价。



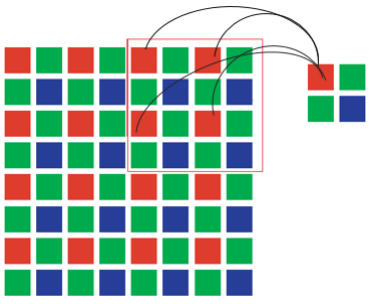
信息发生损失

bin 作为名词常见意思是箱子（比如垃圾箱），作为动词则是指分类装箱，比如在LED生产线上需要把发光参数相近的LED颗粒分装到同一个箱子里，这个操作就叫做binning。在统计图像的直方图时，人们会把像素允许的取值范围划分成若干个bin，则每个像素都会落入某一个bin中。

下图描述了一种对水果进行binning的方法，按照类似的原理，可以把水果替换成任何需要统计和分类处理的东西，比如图像里的像素值。

binning

对于CMOS sensor，binning 是一种特别的工作模式，其特点是NxN个同类像素会合并成一个假想的大像素，合成像素所携带的信息包含了每个原始像素的贡献，所以不会像decimation那样有些线条完全没有被采到。图像的视场角也没有变化。同时，由于输出像素数量变少，所以帧率也可以提高。



2x2 binning

几个像素合并成一个大像素之后，大像素能够拦截的光能量正比于参与合成的小像素数（NxN）。按照比较简化的噪声模型，像素噪声的主要来源是光信号本身的统计涨落，用泊松分布描述，因此像素噪声的数值正比于sqrt(NxN)=N，所以，binning之后像素的信噪比为单个小像素的N倍。更大的信噪比意味着更好的图像质量，在同样的图像亮度下，binning模式的画面会更加干净、通透。

以上是CMOS sensor 所支持的几种降采样方式，不难注意到，decimation和binning 都是支持有限的基于整数的降采样比例，无法支持任意有理数比例的降采样倍数，因此其作用注定是有限的，不足以满足全部的需求场景。

举例来说，数字电视系统的标准分辨率是D1=720x576，如果CMOS sensor 输出的原始分辨率是1280x1024，则该图像需要横向、纵向都缩小1.7777倍才能得到D1分辨率的图像。这个需求无法用decimation或者binning方法实现。在不改变原始分辨率、不改变视场角的前提下，cropping

也无法使用。所以，除了sensor之外，我们还需要有额外的更加通用灵活的机制来解决有理数比例的降采样需求。

不难发现，1280:720=16:9，也就是每16个输入像素应该产生9个输出像素。显然，我们很容易排除一个最简单粗暴的做法，就是保留前9个像素而丢弃后7个像素，这个做法的缺点是显而易见的。更加公平的策略是，每个输入像素的9/16应该贡献出来，成为某个（或某几个）输出像素的一部分，而剩下的7/16则应被丢弃。从这个思路出发可以发展出面积平均法，下面以4:3为例这种降采样方法的工作原理。

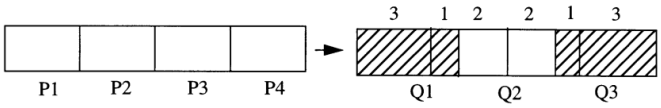
面积平均法

如下图所示，我们需要从4个P像素生成3个Q像素，每个P像素需要严格地取出3/4，丢弃1/4。

Q1从P1中取走3/4后，还缺1/4，于是再从P2中取走1/4，变成一个合格的输出像素。

Q2从P2中取走2/4后，还缺2/4，于是再从P3中取走2/4，变成一个合格的输出像素。

Q3从P3中取走1/4后，再从P4中取走3/4，变成一个合格的输出像素。



$$Q1 = \frac{(3P1 + 1P2)}{4}$$
$$Q2 = \frac{(2P2 + 2P3)}{4}$$
$$Q3 = \frac{(1P3 + 3P4)}{4}$$

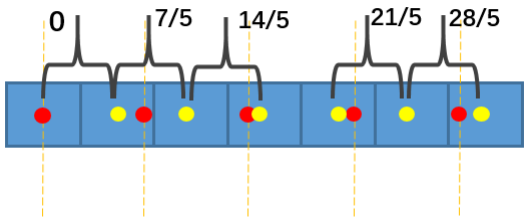
area averaging

扩展到二维时也是类似的思路，读者可自己想象。

线性插值法

线性插值法是一种更加通用的缩放方法，当缩放的倍数在0.5~2.0之间时效果比较好。

下面以7:5缩小为例说明线性插值的具体方法。



线性插值法

图中蓝色为输入像素，黄点为输入像素的中心。我们用红点表示5个输出像素的位置，它们应均匀分布在[0, 6]范围内，两个相邻红点之间的距离为7/5，并且每个红点两侧应存在一个黄点。

由于黄点的位置是已知的，所以我们可以用红点和相邻的两个黄点做线性插值，就可以求得红点处的像素值。

由于这种方法使用的缩放倍数比较有限，当需要更大的缩放倍数时，可以先使用其它方法（比如面积平均法）对图像做预处理。预处理可以预设几个固定的规格，如1:2, 1:4, 1:8, 1:16, 1:32等，剩余的部分可以交给线性插值继续处理。

边缘保持问题

如果单从缩放效果考虑，降采样算法的每个输出像素应该平等地包含NxN个输入像素的信息。然而有时简单的平等也未必就是最好的策略。举例来说，如果待处理的图像中存在一条十分锐利的线条，按照简单平均的策略这个线条会被稀释N倍，于是会变得模糊不清。然而这并不是我们想要的结果，我们认为这个线条属于非常有价值的信息，应分配更大的权重，使其基本完好地保存下来。

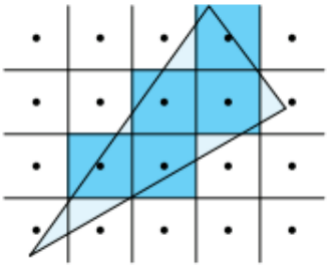
成本考虑

当降采样的缩放系数很大时，比如N=8或N=16，按照上述算法，为了得到一个输出像素，算法需要扫描64个甚至256个输入像素，还要对可能存在的边缘做特殊的处理。这样做真的值得吗？简单的decimation明显会更省成本，要不要折衷一下呢？客户对我们的期望是什么？

锯齿效应

CMOS sensor 输出的图像经过ISP处理后会得到RGB或YUV格式的图像，在对这些图像进行压缩和存储之前，人们可能会希望在图像上加入一些文字、图形、水印等等，这就会涉及在图形上绘制几何图形的操作，也就是2D操作。

前面提到过，CMOS sensor 每个像素都是一个有一定面积的正方形，其实显示器的像素也是类似的正方形。当我们在显示器上画斜线时，斜线经过的正方形像素会被点亮，从而形成如下图所示的台阶或者锯齿。像素尺寸越大则锯齿效应约明显。

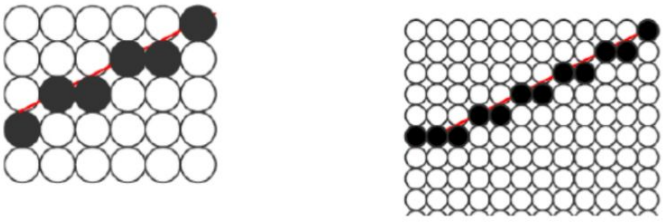


aliasing effect

没有人喜欢锯齿效应，所以凡是需要绘制图形、显示图形的场合都需要考虑如何缓解锯齿效应，相应的方法就是抗锯齿算法。在比较专业的2D/3D应用中（比如游戏行业），人们会广泛使用SSAA和MSAA两种技术实现抗锯齿。

超级采样

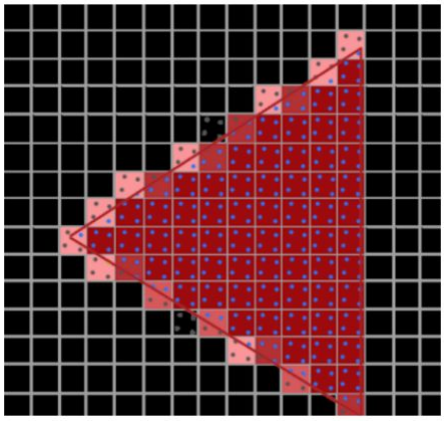
超级采样（Super-Sampling Anti-Aliasing，SSAA）是一种非常有效的抗锯齿方法，它的思想简单粗暴，就是在内存buffer中绘制图形的时候，buffer分辨率是显示分辨率的N倍。



当N=2时，显示器上的每个像素对应buffer中的4个sample，需要计算4次。在最后渲染的时候，这4个sample的值取平均，作为最终显示的值。

多重采样

多重采样（Multi-Sampling Anti-Aliasing，MSAA）是对SSAA方法的一种改进，可以显著地节省计算量，从而节省功耗。MSAA 也是采用每个像素对应4个sample 的方案，但是区别在于，MSAA会评估这4个sample 中有多少个位于三角形的内部。假设评估结果为x个（只可能取0, 1, 2, 3, 4），利用x和已知的背景色(x=0)、前景色(x=4)，可以用线性插值的方法得到一个目标颜色f(x)，将f(x)值统一赋给4个sample，从而省掉了分别计算4个sample的成本。



MSAA

编辑于 2022-07-14 16:35

ISP（互联网服务提供商） 图像信号处理器ISP (Image Signal Processor)