旧方法们：

将图像中的颜色 投影到 与亮度和颜色变化的正交方向上 获得光源不变的图像

在原始图像中存在 但是在（上述得到的光源）不变图像的边界应该是有阴影边界的

归零对应的梯度、重新整合得出一个无阴影的图像 🡪 适用2D

关键点应该由阴影边界两边的有相同反射率的像素确定，因而边界梯度只由光照变化确定

阴影合成公式：I = βL+ (1−β)S

I是一个线性合成的无阴影的L图

S是有阴影的图

β是个变量参数

## Section1：

新方法：

只适用于一个主光源（如户外）

每一个被移除的阴影能够映射到一个场景表面（颜色和纹理一致的区域），这个场景应该有有阴影区和无阴影区。这个阴影应该覆盖额外的相邻区域

阴影移除由用户点击要移除阴影的区域

用户单击原图1a后，计算出代表表面高光区的部分1c、代表阴影区的部分1d、完整阴影区1e

1b：阴影种子（Section2有介绍是什么）

1c：相同表面高光区

1d：相同表面阴影区

1e：计算出的完整阴影遮罩

1f：保存的阴影区内部

1g：完成除去过程

## Section2 检测阶段：

确定要恢复的阴影像素以及其他在恢复阶段会用到的区域

这部分的输出结果有三个：

Ms：确定一个完全在阴影里的区域，其所有像素都属于一个单独的场景表面（有相同色彩和纹理）

Ml：确定一个完全不在阴影里的区域，其所有像素属于和Ms相同的场景表面

Mshadow：确定阴影移除需要的全部像素

前两个输出用于导出阴影结构模型的参数

Ms是Mshadow的子集，可能由一些不同场景的阴影区域组成

### 2.1 阴影种子

阴影检测移除首先要用户单击阴影内部以提供阴影位置区域。而且这个地方所在的场景表面要同时包含阴影和高光区（不能是图1的足球）

然后对用户选择的像素重复进行区域扩展操作，形成一个同阴影场景区域内的小块，这就是阴影种子，图1b

大概执行3-4次区域扩展，每次区域其扩展相邻区域的颜色只能有很小的容差并且不能越过很强的边界

阴影种子的检测由一个过滤、降低采样版本的图像完成，因为这样可以减少对颜色上小变量的敏感度，也能加快处理进程

### 2.2 计算Ms和Ml

接下来将阴影种子延伸成一个完整的Ms遮罩，形成这个场景表面的阴影部分，并计算Ml遮罩用于相同场景表面的光照部分。

我们要先确定图像中和阴影种子属于相同场景的像素，不论是光照部分还是阴影部分，然后扩展这个场景对应的阴影种子，直到其包含整个阴影区域。而这个场景的补集部分就会是光照区域。在这之前，图片将会被一个双边过滤器过滤。也是因为降低敏感度、快速处理的原因。

找到其他和shadow seed相同场景的像素：在光照不变的颜色距离度量下，选择和shadow seed的颜色差别较小的像素。在两个RGB颜色下不变距离的定义是1-|cosθ|，θ是两个颜色对应的3-vectors

RGB颜色下，有着相近反射率的点会和相近的共线向量吻合（代表这些点的向量应该共线？），不论其是在阴影下还是光照下

\*？ 这个Vector是什么东西，以及生成向量？

使用如上距离度量，可以计算每个像素的距离，以及shadow seed的中值。查看这些距离的直方图，可以明显观测到一个很大的像素峰值，其和shadow seed的色差距离几乎是0。因而可以根据这个初始峰值设置一个阈值，所有的色差距离在这个阈值之内的像素都可以被分类为和shadow seed有着一样的场景表面

有两种方法度量两种颜色的相似性，。第一种简单地看Y（luma）通道，因为阴影像素应该和shadow seed有相似的亮度，而亮像素则会更亮。但是也有很多情况非阴影区域也会很暗。在这种情况下有一种更有效的方法来度量相似性，用chroma（Cb、Cr）通道。第二种方法需要实施扩增处理两次，使用两种不同的方法。最终处理引起了最小的Ml和Ms相似内容数值

### 2.3 计算Mshadow

获得完整阴影区域的二值遮罩，使用Alpha matting algorithm（Alpha消光算法）来达到。构造一个trimap，标明了什么一定是在阴影内，什么一定是在阴影外，以及什么是不知道的。这个算法会产生这个图像的一个完整的alpha通道。这个不光滑的阈值用来产生Mshadow二值阴影遮罩。

Trimap的构造方法：从早期处理并通过增强扩展区域获得的Ms区域，但是是属于不同场景表面的（比如那些被Ms包围但是不属于Ml的像素）

所有这些扩增区的像素都一定是在阴影里的。然后膨胀这个区域并求补集来获得的像素被认为一定是区域外的。剩余的像素就被认为是未知的

图2现实了三个遮罩。

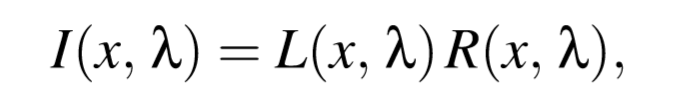
## Section3：

提出了新的基于金字塔的恢复过程，应用于产生无阴影且没有对比度损失或产生噪点的方法

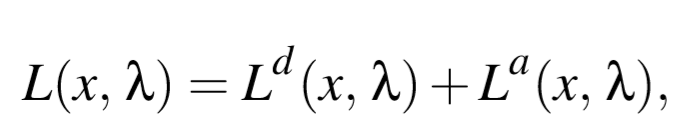
从得到一个（一张图上阴影像素的颜色和无阴影颜色）的仿射关系和开始

这个仿射模型的参数可以由阴影内不同区域来估计，使用之前获得的范围遮罩

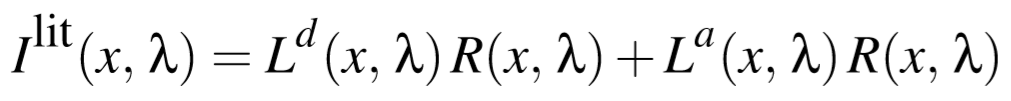
### 3.1 阴影结构模型



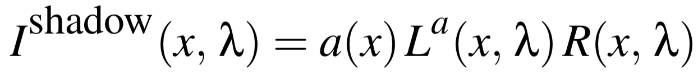
I(x,λ)是在波长λ下，场景中点x的反射强度，L和R是同一个点同一个波长下的光照和反射率

若点x是高光无阴影点，则光照可以表示为

Ld是直接光照，La是剩下的非直接光照（环境光照）



有物体挡住主光源的情况，在点x上投影了阴影

反射强度为

a(x)是空间向量，会导致阴影区域内周围光照减弱

要假设周围的减弱因素是一样的 否则a和λ 也相关

x的光照强度下，阴影强度仿射方程

I lt(x, X) — 
— Ld(x, X) + —1 
shadow 

关联与同场景内点x，点p的真实颜色能通过用传感器的光学反应积分I(x,λ)获得

这种线性关系不会改变阴影和光照的仿射关系

Illt(p) — 
shadow 
(p) （方程7）

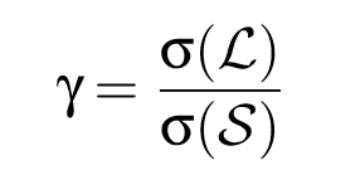
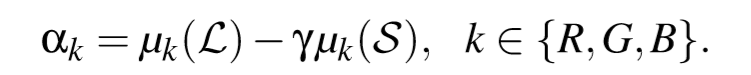
ak，k属于RGB，是相机对直接光照的RGB三通道映射

γ(p) = 1/a(x) 是周围光线减弱因素的倒数，并不和光波长相关。因而使用这四个线性参数，光照像素的颜色可以从其阴影颜色恢复

### 3.2 参数估计

仿射模型中有4个参数，用于把像素从阴影恢复到光照下

估计这4个参数需要两组像素，一组在阴影区域内，一组在阴影区域外。这两组像素可以来自不同场景表面，但是为了能够使参数稳定，最好来自同一场景表面。同时这些像素的颜色拥有和相关Mask一样的出现频率（Ms对应阴影区，Ml对应光照区）。用S和L表示两个像素的结果集。把S和L放到方程7中估计aR、aG、aB和γ。μ(S)和μ(L)表示S和L的平均颜色，σ(S)和σ(L)表示标准差。然后得到

### 3.3 基于金字塔的恢复方法

出现的问题1：恢复的区域对比度较源有亮区低。原因：直接光照在无阴影区域会带来更高的对比度。

问题2：恢复的区域可能呈现更多噪点，尤其当阴影在平滑或者黑暗的表面。

计算Lapcacian金字塔的同时也生成低画质版本的图像

### 3.4 处理不均匀阴影

越接近遮挡物体越暗的情况。因为更多的直射光被挡住了，此时单一一组参数无法完全移除阴影区域

解决：估计另一组仿射参数用于阴影内不同区域。生成一系列阴影内长条采样点，从其内部一直扩展到边界，每一条都生成一组参数。参数计算只用颜色在Ms和Ml遮罩中出现足够多的像素子集。这种处理只在金字塔的粗糙层进行处理。

## Section4 边界处理