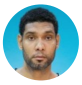
传统图像去雾/增强算法相关论文之暗通道先验



核桃分心木

最近要了解一下图像去雾，图像增强相关的内容。写在知乎这里记录一番。

第一篇必须要看就是He Kaiming的暗通道先验去雾的论文。他的博士论文写的很清晰，简明通透，看着很舒服。

Background:

首先，基于物理模型的图像去雾问题是解决ambiguity的问题，既方程的个数少于未知量的个数。这种情况下会导致多解。举个例子，比如三元一次方程：

$x+y+z=1$ ，

这种情况下是没办法得到 x,y,z 的确定解的。

常见办法有1）增加更多已知的变量->来减少未知量的个数。2）增加额外约束或假设（常称为prior，先验信息）-> 增加方程的个数。

对于图像去雾问题，有雾条件下图像生成模型为：

$$I(x)=J(x)t(x)+(1-t(x))A$$

变量解释： I 是存在雾的影响下，观测到的图像(RGB 3D vector)， J 是无雾条件(haze-free)下的图像(scene radiance)。 $t(x)$ 是transmission map or transparency of the haze. 取值在[0,1]之间， $t(x)=0$ 意味雾完全不透明，全遮挡， $t(x)=1$ 意味完全没有雾。 A 是环境光。

物理意义分析：依据图像生成模型可以看到，图片是由两部分组成，一部分是由于大气对scene radiance的吸收 $J(x)t(x)$,另一部分是环境光的散射 $(1-t(x))A$ 的contribution 。基于这个公式，也可分析出两个term对生成雾的图片质量的影响。

1）通过对生成模型公式两侧求导，可以看到由于 $t(x)$ <1,会导致 $I(x)$ 的contrast(visibility)比原图 $J(x)$ 低。既haze的存在导致图像对比度下降。

2）其次，由于airlight-A的存在，从矢量相加的结果来看，$I(x)$ 和 $J(x)$ 在RGB色彩图像空间中方向不同。因次会导致颜色出现偏差。

问题描述：因此图像去雾问题可以描述为，如何基于 $I(x)$ 求解 $J(x)$ 。

显然，要想求解 $J(x)$ ，需要计算出 $t(x)$ 和 A 。问题是这个是ill-posed/under constraint的问题。我们可以通过比较未知量的个数和方程的个数来判断问题的状态。假设图片像素个数为N。RGB，3个通道，都满足图像生成模型，即方程数为3N。未知数: J(x)，3N; $t(x)$,与颜色通道无关，所以未知量为N； 环境光A三个通道三个value,所以未知量个数为3。 因此总的未知量个数为4N+3。所以，至少要增加N个方程/约束才能求解。

暗通道先验（dark channel prior）：

For outdoor haze-free images, in most patches that do not cover the sky, there exists some pixels whose intensity is very low and close to zero in at least one color channel.

对于室外没有雾条件下拍摄的图片 $J(x)$ ，绝大多数不包括天空的像素块，像素块内一定存在一些像素点，在至少一个颜色通道内，亮度intensity极低，趋于零。这些像素被成为暗像素。暗像素主要出现在在包含以下三种情况的image patch中： 1） 阴影 2）黑色物体 3）有色物体。 前两种由于像素块几乎全部为黑色，容易出现暗像素。第三种有色物体比如绿色，则通常绿色通常的强度很高而其他通道intensity值极低。

数学描述：

$$J(x)\ is\ a\ dark\ pixel \Leftrightarrow \min_{c\in\{r,g,b\}}(J(x)) < \delta$$

对于image patch，至少包含一个暗像素的充要条件：(这里就开始用0 代替 small value δ 了)

$$\min_{x^{'}\in\Omega}(\min_{c\in\{r,g,b\}} J(x^{'})) <= \delta \approx 0$$

Dark Channel 暗通道定义：

$$J^{dark}(x) = \min_{x'\in\Omega(x)}(\min_{c\in\{r,g,b\}} J(x'))$$

Dark Channel Prior 暗通道先验信息（统计了大量网络图片得到的结论）：

$$J^{dark}(x) = \min_{x'\in\Omega(x)}(\min_{c\in\{r,g,b\}} J(x')) \approx 0$$

接下来，基于暗通道先验，开始疯狂的表演：同时做dark channel计算(两侧同时取两次最小值。)

$I(x)=J(x)t(x)+A(1-t(x))$

对每一个color channel c：

$$\min_{x'\in\Omega(x)}\min_c\frac{I_c(x')}{A_c}=\min_{x'\in\Omega(x)}\min_c(t(x')\frac{J_c(x')}{A_c}+1-t(x'))$$

假定每个patch中 $t(x)$ 是uniform的，表示为 $\hat{t}\,(x)$

$$\min_{x'\in\Omega(x)}\min_c\frac{I_c(x')}{A_c}=t(\hat{x})\min_{x'\in\Omega(x)}\min_c(\frac{J(x')}{A_c})+1-\hat{t}\,(x)$$ ，

Thus,利用暗通道先验为0，可得transmission map：

$$\hat{t}\,(x)=1-\min_{x'\in\Omega(x)}\min_c\frac{I_c(x')}{A_c}=1-\frac{I^{dark}}{A}$$

估计A:

假设所有color channel A相同， $A=A_c=A_b=A_g$ 。无论A的大小，一定有 I^{dark} 越大，$t(x')$ 越小。当 $t(x')\simeq 0$ 时，此时一定有 $1-\frac{I^{dark}}{A}=0$,因次，可以用 I^{dark} 的最大值来近似 A 。

因此，选择top 0.1% brightest pixels in dark channel。

Scene Recovery:

$$J_c(x)=\frac{I_c(x)-A_c}{\max(t(x),t_0)}+A_c$$

分母 t_0 避免奇异，通常取0.1。

至此，基本版的暗通道先验就算完成了。kaiming用了一个简单的近似，实现了超牛的效果。

一些优化：

1）亮度调节：

去雾操作会使得图片变暗，因次将图片乘以一个系数C,使得J(x)与I(x)的平均亮度相同。

2）Aerial Perspective：

有时，并不能把雾全部除去。比如在有雾的情况下，远处的山会显得更白一些。人会依据这中条件，来增加我们对距离的判断。因次，对于远处的物体，我们要保留一些雾。

$t(x) := 1 - \kappa(1 - t(x))$

κ 通常取值为0.95。这里注意分析这个公式的作用：公式本身就是线性的。所以把 $t(x) = 0, t(x) = 1$ 分别带进去看一下效果就可以了。 $t(x) = 0$ 时，表示完全不透明(远处物体），此时通过新的赋值公式，给 $t(x)$ 保留一些value，实现了初衷。当 $t(x) = 1$ 时，表示完全透明，此时新的赋值公式无影响。

3）对 $t(x)$ 的优化：

这个是更重要的优化过程。基于上面得到的 $t(x)$ 过于粗糙，希望可以refine原始的 $t(x)$ 。refine的办法是基于，基于初始得到的 $t(x)$ ，重新估计 $t(\hat{x})$。使得在有深度突变的时候， $t(\hat{x})$ 在有变化表现的同时，空间上保持平滑。这种既有pixel-wise constraint 又有spatial constraint的问题，比如立体视觉，图片降噪等等，常用的解决方案就是MRF，建模如下：

$$E(t) = \lambda \sum_x ||t(x) - t(\hat{x})||_2^2 + \sum_x \sum_{x' \in N(x)} w(I, x', x)||t(x) - t'(x)||_2^2$$

第一项data term， 第二项，smoothness term主要通过weight $w(I, x', x)$ 来控制。当图片 I 存在sharp edge时（近似depth discontinuity), 令 $w(I, x', x)$ small value，减轻smooth的效果，保留discontinuity的细节。

把优化问题转到矩阵形式。并且利用image matting的求解办法来求解。

$E(t) = \lambda||t - \hat{(t)}||_2^2 + t^T L t$

这里矩阵L是matting Laplacian matrix。可以利用的原因，image matting的数学表示

$I(x) = F(x)\alpha(x) + B(x)(1 - \alpha(x))$ 和有雾图像生成模型很相似。 $t(x)$ 和 $\alpha(x)$ 的作用效果类似。在image matting的问题上，Laplacian matrix可以很好的recover出 $\alpha(x)$ 。因次kaiming站在巨人的肩膀上。矩阵元素的计算方法，以及线性矩阵的求解detail不表，因为后面发现，这个算法实在太慢了。

Overall,整个算法的流程：

a) 估计环境光 A

b)求解transmission map $t(x)$

c) refine $t(x)$ 得到 $\hat{t}(x)$

d) 增加aerial perspective κ

e) scene recovery $J(x)$

算法的局限性：

两点局限性都是来自数学假设不满足时：

1) dark channel prior 条件不满足时，比如大块区域很白，很亮，不存在dark pixel时。

2）haze image equation 不满足时，比如 $t(x)$ 与color channel有关；存在point light sources，导致环境光 A non constant value时。

另一方面在运算效率上，soft matting比较慢，kaiming的改进算法，guided filter（翻译为导向滤波）的方法在第二篇中介绍。

发布于 2018-05-13