单幅图像基于暗通道先验知识的去雾算法

Dezeming Family

2023 年 2 月 17 日

DezemingFamily 系列文章和电子书**全部都有免费公开的电子版**，可以很方便地进行修改和重新发布。如果您获得了 DezemingFamily 的系列电子书，可以从我们的网站 [https://dezeming.top/] 找到最新的版本。对文章的内容建议和出现的错误也欢迎在网站留言。

目录

一　引言

我们参考的代码来自于 [2]，我增加了一些注释，放在了 [3] 中。
整个代码流程分为：

- 求出暗通道
- 计算大气光强
- 估计 $\widetilde{\mathbf{t}}$ 并将其作为约束，使用引导滤波细化得到 $\mathbf{t}$
- 恢复场景 radiance

我们分为四节来介绍，由于源码中注释已经非常详细了，所以我们描述会简单不少。

二　求出暗通道

DarkChannel() 函数先求出每个像素最暗的通道颜色。然后使用腐蚀方法，腐蚀是计算一个窗内的最低值作为当前窗像素中心的值。

三　计算大气光强

AtmLight() 函数首先选择的其中暗通道最亮的百分之 0.1 比例的像素；然后把暗通道图和原图都拉成一个数组（把长宽这两个维度拉成一维）；之后对暗通道图的元素按照亮度进行排序，获得排序序号；把最亮的百分之 0.1 的像素作为对大气光 A 的估计，把它们求平均，就得到了 A。

四　估计 $\widetilde{\mathbf{t}}$ 并将其作为约束，使用引导滤波细化得到 $\mathbf{t}$

该过程其实就是论文原文的公式 (12)：

$$\tilde{t}(\mathbf{x}) = 1 - \omega \min_{\mathbf{y}\in\Omega(\mathbf{x})} \left(\min_{c} \frac{I^c(\mathbf{y})}{A^c} \right). \quad (12)$$

TransmissionEstimate() 实现时有一个巧妙之处在于，求两次 min 相当于再调用一次 DarkChannel() 函数。DarkChannel() 函数的输入是 $\frac{I^c(\mathbf{y})}{A^c}$。

五　恢复场景 radiance

恢复的过程 Recover() 函数就是论文原文的公式 (22)：

$$\mathbf{J}(\mathbf{x}) = \frac{\mathbf{I}(\mathbf{x}) - \mathbf{A}}{\max(t(\mathbf{x}), t_0)} + \mathbf{A}. \quad (22)$$

那么整个图像去雾算法流程就结束了。

参考文献

[1] He, Kaiming, Jian Sun, and Xiaoou Tang. "Single image haze removal using dark channel prior." IEEE transactions on pattern analysis and machine intelligence 33.12 (2010): 2341-2353.

[2] https://github.com/He-Zhang/image_dehaze

[3] https://github.com/feimos32/ComputerVision-Code-Implementation-and-Collection