

数字图像处理

大作业三

图像去雾

学号 2017011589

姓名 吾尔开西

班级 自 76

目录

- 一、总述.....3
- 二、文献调研4
- 三、暗原色先验算法5
 - 1、基本模型5
 - 2、暗原色先验6
 - 3、估算透射率6
 - 4、估算全局背景光8
 - 5、细化透射率9
 - 6、整体思路11
- 四、暗原色先验算法的改进11
 - 1、全局背景光计算的改进11
 - 2、浓雾部分的改进13
 - 3、直方图均衡18
 - 4、算法加速19
- 五、AOD 神经网络去雾.....20

1、问题建模	20
2、网络模型	21
3、训练数据	22
4、改进	23
六、结果	24
七、总结	28
八、参考资料	29

一、总述

本次大作业要求调研实现去雾算法，发现其中的问题，并对算法进行改进。

我首先实现了基于暗原色先验的去雾算法，并从运算速度和去雾效果方面进行了一定的改进。

之后，我训练了 AOD 卷积网络来进行图像去雾，并对数据集图片做一定的处理，增加了网络的鲁棒性，去雾效果也很不错。

暗原色先验的去雾算法使用 MATLAB 实现，使用 MATLAB 的 GUI 设计了用户界面；AOD 卷积网络使用 Python 实现，使用 PyQt 设计了用户界面。

二、文献调研

室外图像的质量有时会因为雾和霾的原因有所下降，空气中的灰尘和水气对光线的吸收和散射，使得进入摄像机镜头的光混杂了白光，从而导致实际景象的对比度降低。

去雾算法一直受到研究者的关注，传统的去雾算法有多图片去雾和单图片去雾两大类。[Shwartz, Namer, and Schechner \(2006\)](#)利用多张不同角度拍摄图片的信息来实现去雾，属于多图片去雾算法。在深度神经网络广泛应用之前，单图片去雾的算法已经取得了一定成就。[Tan \(2008\)](#)发现无雾图像一定比有雾图像对比度更高，他使用最大化局部对比度的方法实现去雾，但这种方法可能带来色彩的失真。[Fattal \(2008\)](#)通过估计场景的反射率的方法去雾，他假定光传播与表面纹理局部无关。他的方法能较好的保留景象的色彩，但对浓雾的处理效果较差。

2011 年提出的暗原色先验方法([K. He, Sun, & Tang, 2011](#))引起了学界的广泛关注，但方法基于暗原色假设，使用简洁明了的公式处理有雾图像，在色彩不失真的前提下有效地去除雾气。本文中使用的第一种算法也是基于暗原色假设。

近年来，深度神经网络得到广泛应用，在图像去雾领域也不例外。[Ren et al. \(2016\)](#)使用多尺度的网络进行去雾，首先使用训练一个网络来预测图片整体的传递函数，再用一个网络进行细化，与传统的去雾流程有些相似。2017 年提出 AOD-net 使用端到端的网络实现去雾([Li, Peng, Wang, Xu, & Feng, 2017](#))，网络结构精巧，参数量少，但去雾效果很好。与之前的方法不同，AOD 直接将原图输入模型得到去雾后的图像，这有利于直接训练模型来减少结果图像与 ground truth 的误差。本文使用的第二种方法便是基于 AOD。

三、暗原色先验算法

1、基本模型

去雾算法广泛使用的模型如下：

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

其中， $I(x)$ 是有雾图像， $J(x)$ 是无雾图像， $t(x)$ 是透射率， A 是全局背景光。如果估算出了 $t(x)$ 和 A ，那么无雾图片就能用下面的公式计算出来：

$$J(x) = \frac{I(x) - A}{t(x)} + A$$

实际应用，为了防止 $t(x)$ 的值过小，会引入参数 $t_{threshold}$

$$J(x) = \frac{I(x) - A}{\max(t(x), t_{threshold})} + A$$

因此，去雾算法的任务就变成了估计全局背景光 A 和透射率 $t(x)$ 。

2、暗原色先验

暗原色先验（或暗通道先验），是说对于无雾图像中的一小片区域，其三维矩阵的最小值总是很低，换句话说，该区域 RGB 通道的最小值中的最小值总是很低。设图像的暗通道为 $J^{dark}(x)$

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

$\Omega(x)$ 是 x 领域的一小片方形区域。暗原色先验是假设 $J^{dark}(x) \approx 0$ ，但这对天空区域是不适用的。

3、估算透射率

利用暗原色先验假设，我们可以估算出一个粗略的透射率

我们先对 $I(x) = J(x)t(x) + A(1 - t(x))$ 两边除以 A

$$\frac{I(x)}{A} = t(x) \frac{J(x)}{A} + 1 - t(x)$$

求两边的暗通道：

$$\min_{y \in \Omega(x)} \left(\min_{c \in \{rgb\}} \frac{I(x)}{A} \right) = \min_{y \in \Omega(x)} \left(\min_{c \in \{rgb\}} t(x) \frac{J(x)}{A} + 1 - t(x) \right)$$

假设 $t(x)$ 在小片区域内是不变的，并设其为 $t^{\sim}(x)$:

$$\min_{y \in \Omega(x)} \left(\min_{c \in \{rgb\}} \frac{I(x)}{A} \right) = t^{\sim}(x) \min_{y \in \Omega(x)} \left(\min_{c \in \{rgb\}} \frac{J(x)}{A} \right) + 1 - t^{\sim}(x)$$

A 在整个图像范围内是常数，由暗原色先验假设

$$\min_{y \in \Omega(x)} \left(\min_{c \in \{rgb\}} \frac{I(x)}{A} \right) \approx 0, \text{ 因此:}$$

$$\min_{y \in \Omega(x)} \left(\min_{c \in \{rgb\}} \frac{I(x)}{A} \right) = 1 - t^{\sim}(x)$$

于是我们就计算出了一个粗略的透射率 $t^{\sim}(x) = 1 -$

$$w \min_{y \in \Omega(x)} \left(\min_{c \in \{rgb\}} \frac{I(x)}{A} \right)$$

其中，引入参数 w 是为了保留一定程度的雾，以使去雾结果更加自然。下面是本任务的图片计算出来的 $t^{\sim}(x)$

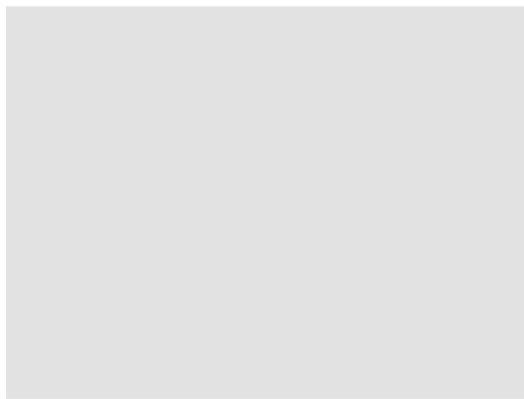


4、估算全局背景光

一般的方法中，估算全局背景光是通过寻找局部极值点的方式来进行，但这种方式可能会造成误判，比如图片中白色的物体（本图中的鹅身）会带来干扰。

暗原色先验方法中，估算全局背景光用到了图片的暗通道值，取出图像暗通道值排前 0.1% 的像素坐标，再取这些坐标对应的原图中亮度值最高的亮度，作为全局背景光的值。在一张图片中全局背景光是常数，为计算方便，把它展成和图片一样大小的矩阵。

用这种方式得到的结果鲁棒性较强，因为非浓雾区域暗通道值很低，所以得到全局背景光点落在白色物体上的概率很小。下面是本任务图片中的全局背景光展成矩阵的结果。



5、细化透射率

用 3 中的方法得到的透射率 $t^{\sim}(x)$ 只是较为粗略的透射率，为了提高去雾质量，需要对其进行细化。这就用到了同一作者另一篇文章里的引导滤波算法([K. He, Sun, & Tang, 2013](#))

设算法的输入图像为 p ，引导图像为 I ，输出结果为 q 。在本任务中， p 为粗略的透射率 $t^{\sim}(x)$ ， I 为原图像， q 为精细化的透射率 $t(x)$ 。有如下的关系：

$$q_i = a_k I_i + b_k, \forall i \in w_k$$

其中， a_k ， b_k 是滤波参数， w_k 是图像中以 i 为中心的一小片区域。

滤波参数的计算方法如下：

$$a_k = \frac{\frac{1}{|w|} \sum_{i \in w_k} I_i p_i - u_k \bar{p}_k}{\sigma_k^2 + \epsilon}$$

$$b_k = \bar{p}_k - a_k u_k$$

其中， u_k 是 l 中区域 w_k 的均值和方差。

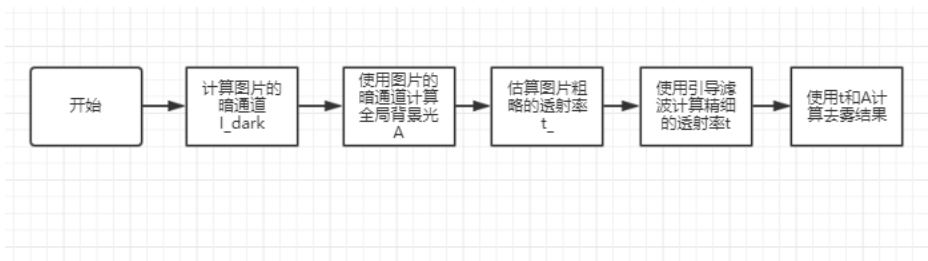
得到滤波参数后便可计算输出 $q_i = a_k I_i + b_k$ ，下面是本任务图片的粗略透射率 $t \sim(x)$ 用 rgb 原图引导滤波后的结果：



可以看出，精细化的透射率与 $t \sim(x)$ 相比少了很多毛刺，但是也可以看到图片浓雾部分的透射率变得与其他部分几乎没有区别，这可能导致浓雾无法去除，我们将在下一部分详细讨论。

6、整体思路

整个算法的整体思路也很清楚，用到了上面说的几步，流程图如下：



四、暗原色先验算法的改进

1、全局背景光计算的改进

在原算法中全局背景光的计算已在上一节中详述，大体上是通过求条件极值的方式获得全局背景光。我们知道暗原色先验假设只对图片的非天空区域成立，但原作者称这种求取方式适用于有天空部分的图片。

实际上，这种方式并不是对所有情况都完全适用，比如对下面这张图片的去雾：



使用原算法去雾的结果如下，可以看出天空部分有明显的色彩畸变，景物部分整体偏暗。



原算法求得的背景全局光为[0.9865, 1, 1]，值过大，因此我们认为是背景全局光的估算出了问题。

原算法中使用条件极值求解背景全局光，我们改为使用条件均值，即对原图像暗通道值排前 0.1% 的像素求均值。另外，对全局背景光的上限做约束，使其不会过大。修改后得到的去雾图像如下，天空部分的色彩畸变消失，景物部分亮度提高。



2、浓雾部分的改进

暗原色先验方法有时对图像浓雾部分的处理效果不佳，如下图红圈部分，基本与原图相同，没有去雾效果。



我们在上一节中已经说明结果图像的计算公式：

$$J(x) = \frac{I(x) - A}{\max(t(x), t_{threshold})} + A$$

其中，浓雾部分的透射率 $t(x)$ 应偏小。我们知道 $t(x)$ 是由粗估的透射率 $\tilde{t}(x)$ 引导滤波得到的，观察一下粗估的透射率 $\tilde{t}(x)$ ，如下图：



浓雾部分（红圈内）偏黑，粗估的透射率 $\tilde{t}(x)$ 确实偏小。我们再观察一下精细化的透射率 $t(x)$ 如下图。浓雾部分的透射率基本与周围区域相同。因此可以判断，浓雾部分去雾效果差是由于引导滤波过程中，浓雾部分的透射率偏高。



为了证实这一结论，我们用粗估的透射率 $\tilde{t}(x)$ 代替精细的透射率 $t(x)$ 计算去雾图像，如下图所示。虽然其他景物部分恢复效果很差，但浓雾部分的去雾效果明显好于原算法。



如何结合精细透射率和粗估透射率的优势？我们决定利用粗估透射率的值对精细透射率的值进行校正。

拿出粗估透射率中值偏低的部分（浓雾部分），对精细透射率中对应的像素点乘一个小于 1 的系数 k_{correct} ，其他部分不变。为了使修正后的透射率保持连续性，我们可以对系数矩阵 k_{correct} 做均值滤波。利用修正后的透射率计算的去雾图像如下，浓雾部分的去雾效果优于原算法，而其他部分不受影响。



3、直方图均衡

为了增强去雾的效果，可以对算法的结果使用直方图均衡算法，但直方图均衡会使图像的色彩产生一定程度的失真，所以只在 UI 中作为可选项提供给用户，来满足用户不同的需求，下面是对结果图片做直方图均衡后的效果，可以看出去雾的性能确实有所提升。



4、算法加速

使用快速引导滤波算法进行加速([Kaiming He & Sun, 2015](#))。在用引导滤波算法计算精细的透射率 $t(x)$ 时，需要用如下公式求得系数

$$a_k = \frac{\frac{1}{|w|} \sum_{i \in w_k} I_i p_i - u_k \bar{p}_k}{\sigma_k^2 + \epsilon}$$

$$b_k = \bar{p}_k - a_k u_k$$

实际上，系数 a ， b 的求解不需要用到 I 和 p 的细节信息，因此我们可以先对 I ， p 做降采样，使用降采样的结果求解 a ， b ，再将求得的 a ， b 矩阵扩大为原图尺寸，能得到和原算法基本相同的效果，而计算速度快了 k 倍（降采样倍数）。

五、AOD 神经网络去雾

卷积神经具有很强的特征提取和学习能力，CNN 网络进行视觉任务不需要人为提取特征，网络的参数学习过程便是提取和筛选特征的过程。CNN 的模型被提出以来便被应用到各个计算机视觉任务中，并取得惊人的成绩，比如图像分类，手势识别，人脸识别等等，在图像去雾中当然也有应用。我认为在当前的背景下，介绍图像去雾算法必然不能遗漏神经网络的应用，而且实验室也有几块闲置的显卡，于是训练了一个 AOD 卷积神经网络，得到较好的图像去雾效果。

1、问题建模

在本文第三部分已经说明，传统方法中对有雾图像的建模方式用如下公式表示：

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

其中， $I(x)$ 是有雾图像， $J(x)$ 是无雾图像， $t(x)$ 是透射率， A 是全局背景光。

使用神经网络的各个方法其实也是用这个公式进行建模，在 AOD 被提出前，研究者通常别计算 $t(x)$ 和 A ，再用这两个矩阵计算 $J(x)$ ([Ren et al., 2016](#))。这就带来一个问题，当我们训练神经网络时，需要定义一个损失函数，来表征模型的预测能力并惩罚模型（梯度后传），如果我们要训练一个模型预测图像的透射率 $t(x)$ ，必须知道精确的 $t(x)$ 才能准确计算损失

函数，从而训练出一个不错的网络。但问题是图像的透射率往往是未知的。AOD 网络使用一个巧妙的方法解决了这个问题。

将上面的传统公式进行转化：

$$J(x) = \frac{1}{t(x)} I(x) - A \frac{1}{t(x)} + A$$

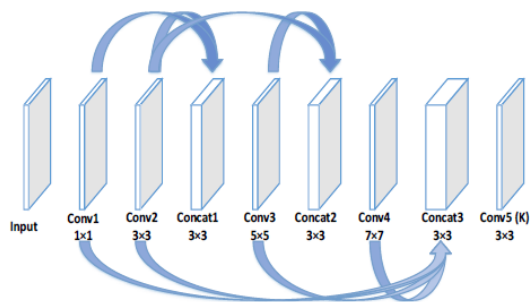
再设 $K(x) = \frac{\frac{1}{t(x)}(I(x)-A)+(A-b)}{I(x)-1}$ ，上面的公式就变成了：

$$J(x) = K(x)I(x) - K(x) + b$$

其中，**b** 是一个常数。因此，我们只需训练一个网络从原图像求出 $K(x)$ ，再用上面的公式求出 $J(x)$ ，便实现了一个端到端的网络。

2、网络模型

网络模型如下图所示，只有 5 个卷积层，卷积层通道不超过 12，参数量很小，是一个轻量级网络。图中的箭头表示两个层的 feature map 拼接得到下一个 feature map。



(b) A closer look of the K -estimation module

3、训练数据

训练数据来自 NYU2 图片集中的 1400 多张室内图片，向图片加入不同程度大小的大气光来模拟雾气效果，将原图作为 ground truth([Li et al., 2017](#))。



4、改进

对该方法的改进可以有修改网络模型、数据增强、调整参数等，但网络模型和参数与本课程内容无关，所以我们从数据增强下手，进行改进。

在传统方法中，图像中的噪声会对输出结果造成影响。这是因为传统方法对全局背景光的估计依赖于图像极值，而椒盐噪声会对极值的求取造成影响。

为了使 AOD 训练出来的模型有较强的鲁棒性，能抵抗噪声的影响，我们在训练图片中加入少量椒盐噪声，如下图所示。



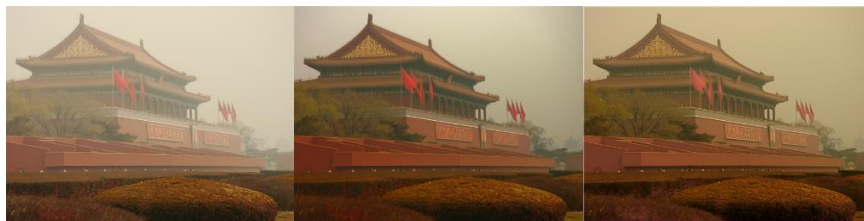
(原图)



(加入椒盐噪声后的图片)

六、结果







(图左：原图；图中：暗原色先验法结果；图右：AOD 网络结果)

上面是两种方法处理得到的一些结果，可以看出在一些图片上暗原色先验法去雾效果比 AOD 网络强（如第一张图片），但 AOD 网络整体的鲁棒性更强，特别是在有天空部分的图片中（如倒数第三张，倒数第二

张图片)，且 AOD 网络的方法得到结果色彩更自然一些（如最后一张图片）。

此外，我分别用 MATLAB GUI（暗原色先验）和 Python（AOD 网络）的 pyqt 制作了用户界面 UI，如下图：





七、总结

本次大作业开始动手写代码前进行了较为充分的文献调研，对图像去雾领域有了一定的了解。在完成暗原色先验算法时，虽然遇到了一些困难，但还是顺利解决，比如进行快速引导滤波时，如何将彩色图像作为引导图的问题。

在进行暗原色先验算法的改进时，体验到了深入研究乐趣。为了对算法进行改进，不仅要观察结果图像中的不足，还要从中间过程寻找原因，对症下药。

AOD 模型的实现让我感受到深度神经网络的神奇之处，无需人工编写复杂的算法，也不用人工提取特征，神经网络就能以不错的性能完成任务。

总之，经过本次大作业的推进，我对图像去雾领域有了一定的了解，完成大作业的过程中也提高了我发现问题、解决问题的能力，受益匪浅。

八、参考资料

- Fattal, R. (2008). *Single image dehazing*. Paper presented at the ACM SIGGRAPH 2008 papers, Los Angeles, California.
- He, K., Sun, J., & Tang, X. (2011). Single Image Haze Removal Using Dark Channel Prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(12), 2341-2353. doi:10.1109/TPAMI.2010.168
- He, K., Sun, J., & Tang, X. (2013). Guided Image Filtering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(6), 1397-1409. doi:10.1109/TPAMI.2012.213
- He, K., & Sun, J. J. a. p. a. (2015). Fast guided filter.
- Li, B., Peng, X., Wang, Z., Xu, J., & Feng, D. (2017). *Aod-net: All-in-one dehazing network*. Paper presented at the Proceedings of the IEEE International Conference on Computer Vision.

- Masnou, S., & Morel, J. (1998, 7-7 Oct. 1998). *Level lines based disocclusion*. Paper presented at the Proceedings 1998 International Conference on Image Processing. ICIP98 (Cat. No.98CB36269).
- Ren, W., Liu, S., Zhang, H., Pan, J., Cao, X., & Yang, M.-H. (2016, 2016//). *Single Image Dehazing via Multi-scale Convolutional Neural Networks*. Paper presented at the Computer Vision – ECCV 2016, Cham.
- Shwartz, S., Namer, E., & Schechner, Y. Y. (2006, 17-22 June 2006). *Blind Haze Separation*. Paper presented at the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06).
- Tan, R. (2008). *Visibility in bad weather from a single image*.