|  |
| --- |
| **数字图象处理课程作业实验报告**  **阴影检测与去除**  14307130167 蒋骐泽 |

目录

[概述 1](#_Toc483244636)

[阴影检测算法 1](#_Toc483244637)

[阴影区域的特征 1](#_Toc483244638)

[基于自然光模型的一个阴影检测算法 1](#_Toc483244639)

[对于上文阴影检测算法的改造 2](#_Toc483244640)

[算法结果 2](#_Toc483244641)

[阴影去除算法 3](#_Toc483244642)

[阴影去除的难点 3](#_Toc483244643)

[阴影去除算法 3](#_Toc483244644)

[算法结果 3](#_Toc483244645)

[实验效果与对比 4](#_Toc483244646)

[阴影检测 4](#_Toc483244647)

[阴影去除 5](#_Toc483244648)

[总结 6](#_Toc483244649)

# 概述

数字图像中阴影是普遍存在的，而且其为数字图像处理的很多任务，如图像特征提取，图像识别，图像分割带来了不利的影响。一个有效的阴影检测与去除方法可以为接下来的图像处理带来很多便利。

# 阴影检测算法

阴影区域的特征

与同表面非阴影区域相比，图像中阴影区域一般会具有以下特征：其亮度会明显比非阴影区域低；与非阴影区域有分界，界线宽度一般不大，在界线上存在渐变；阴影区域的颜色通道比例和非阴影区域比较接近。我们可以利用这些特征来完成阴影检测的工作。

基于自然光模型的一个阴影检测算法

在白天的户外环境中，主要存在两种光源：太阳光的直射，和天空的光线散射。非阴影区域同时受到了太阳光的直射和天空的光线散射，而阴影区域则没有太阳光的直射，只存在天空的光线散射。显然，太阳光的直射强度要大大高于天空的光线散射强度。对于阴影区域，它和阳光直射区域的差即为太阳直射的颜色分布。如果能够将太阳直射下颜色在三个通道的构成比例计算出来的话就可以很好的辅助进行阴影检测。田建东等[1]通过计算后得出，阳光直射在RGB三个通道上的强度比例为。利用这个特征，就可以较为准确的识别出阴影区域。同时，他们提出，将图像进行如下公式的变换后，同表面的阴影和非阴影区域的结果会很相近：

此时利用一些图像区域分割算法即可将同一表面的内容切割到一起，然后即可进行阴影去除。其完整算法如下：

1. 将原图按照上式进行变换得到新图。
2. 使用分水岭算法在图中分割区域。
3. 对于每个区域，计算其区域内所有点的颜色均值。颜色均值指的是三个通道分别求其均值。
4. 假设在三个通道上均比均值亮的区域为非阴影区域。求出非阴影区域的颜色均值，同时乘上强度比例，这个表面的性质。
5. 为了突出差异，找出表面性质中差距最大的两个颜色通道。令每个像素的特征 为这两个颜色通道的值相减。
6. 取为的平均值。将满足的像素都作为阴影像素
7. 验证阴影的合法性：我们求出阴影区域和非阴影区域的颜色均值并作差。其颜色应该满足这个表面的性质L。考虑到在不同地域、不同时间、不同相机拍摄所得到的照片在强度比例上可能有一定偏差，我们认为内均为合法阴影。
8. 如果目前的阴影是合法的，那么再将所有区域内亮度比颜色均值低的点加入阴影集合。

对于上文阴影检测算法的改造

上文提到的阴影检测算法提出了一个很有用的户外环境光模型。在实践中，这个算法没有起到很好的效果，出现了以下问题：

**用于分水岭算法的图效果不佳，难以将阴影和非阴影合在一起**

算法的根本是要将属于同个表面的阴影和非阴影区域同时提取出来，也不能包含其他区域，否则就会对表面性质的预测和提取阴影、非阴影区域产生很大的干扰。但是实际上有很多情况下运用这个方法并不能很好的让同一表面的区域融合，而且很多时候也会出现两个不同表面在图中值相近的情况。当两个表面被当成同一区域时，这两个表面上的阴影几乎不能检出，同时很容易出现将其中一个表面完全当做阴影的情况。

**选取阴影可能不满足“阴影较暗”的常识**

由于选取阴影时使用的判定条件不涉及阴影颜色数值的绝对大小，而是两个颜色通道的差值，因此可能会出现虽然像素亮度较低，但是差值却比高亮度像素还要高的情况，导致高亮度像素反而被当作了阴影。之后在接下来的处理中，甚至会出现整个区域内的点都会被当作阴影的情况。

由于存在上述问题，我们对上文的阴影检测算法进行了如下改造，来减少这些问题造成的影响。改造算法如下：

1. 不再使用图及分水岭算法。考虑到需要去除的阴影在图像中的面积一般较大，而且阴影和非阴影区域会有比较明显的亮度差异，因此使用简单的正方形分块检测办法，将原图像切割成的块，取灰度值，并将其与的全1矩阵作卷积减少噪点影响。由于阴影和非阴影区域亮度差距大，所以需要满足这个块中最亮点和最暗点差距大于40。同时，该区域应该存在较多点接近最亮点与最暗点，因此认为在归一化后该区域的标准差应该大于0.25。只有满足以上要求的块才继续进行接下来的检测。
2. 类似上文方法，选出非阴影区域，估计表面性质。在选取阴影像素时，需要满足选出来的像素不能已经被作为非阴影区域使用，避免将过亮的点判断为阴影。如果候选阴影通过了上文的检测，那么就将当前块放入总阴影候选中。由于之后会进行延伸，所以省略上文的步骤8。
3. 由于阴影较暗，因此取总阴影候选的亮度平均值，并丢弃亮度高于此平均值的候选。剩余在总阴影候选的即认为是阴影，并将从这些阴影开始扩展，查找完整阴影。
4. 对于每一个属于阴影的块，以这个块的点开始做FloodFill算法。FloodFill采用以SeedPoint的颜色作为全部的基准的方式，最大差阈值则设为该块在进行上文算法中非阴影区域和阴影区域平均亮度的平均。
5. 在FloodFill中，阴影图像可能会存在大量的黑白噪点。因此将图中连通块大小小于1000的连通块颜色反转来消除它们。至此完成阴影区域的检测。

算法结果

进行改造后的阴影检测算法对于大部分阴影均能够准确并完整的找出。但是该算法也存在一些问题：在面对一些颜色特别复杂、丰富的场景时，容易将表面交界处错误识别；在绿地场景等，即某些通道颜色强度大大高于其他通道时，无法通过自然光模型验证而无法检出阴影；图中存在多个亮度差距较大的表面，且均存在阴影时，会舍弃亮表面的阴影等。详细内容会在实验效果与对比一节展示。

# 阴影去除算法

阴影去除的难点

在找出阴影区域后，就需要对阴影进行去除。在进行阴影去除时，需要解决这些难点：

1. 不同的阴影所在表面有着不同的性质，不能简单的通过调整亮度、对比度的方式完成。
2. 同一个阴影中，由于光源不为完美的点光源，阴影边界会存在轻微模糊，影响去除。
3. 在一些较高较宽物体的阴影中，由于不但太阳直射光被遮挡，天空的光线散射也被部分遮挡，使得同一个阴影的不同位置阴影的深度不相同。
4. 阴影检测算法可能不够完美，一些阴影的边缘未被当做阴影的一部分

我们的阴影去除算法需要考虑到阴影去除的这些难点，并尽可能的去除这些难点所带来的对结果的影响。

阴影去除算法

傻逼算法

算法结果

一塌糊涂

# 实验效果与对比

我选择了 NAIVE数据集对各种算法进行了测试。

阴影检测

一片白色

阴影去除

一塌糊涂

# 总结

玩尼玛嗨