# 实验一 基于直方图均衡化、直方图规定化及暗通道先验算法的彩色图像去雾实验

# 实验报告

## 一、实验目的

1. 掌握Matlab库函数Histeq的使用方法，通过调用该函数完成基于直方图均衡化(Histogram Equation)、直方图规定化(Histogram Specification)的图像去雾(Image Dehazing)任务。

2. 了解基于暗通道先验[1] (Dark Channel Prior)的图像去雾算法，能够配置该算法超参数、调用该算法完成图像去雾任务。

3. 了解均方根误差(Root Mean Squared Error, RMSE)，峰值信噪比(Peak Signal-to-Noise Ratio, PSNR)，和结构相似度(Structural Similarity Index, SSIM)等客观图像质量评价指标，能够同时从客观和主观两种评价层面上评估不同图像去雾算法的优劣。

## 二、实验任务

给定4种有雾(Hazy)、无雾(Ground Truth)图像对，如图1所示。分别使用直方图均衡化（表1中简写为HistEq）、直方图规定化（表1中简写为HistSp，建议使用无雾原图做为规定图像）、及暗通道先验算法（表1中简写为Dark Channel），对图1中的4种图像对进行图像去雾实验。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| 1. Forest | |  | 1. NH50 | |
|  |  |  |  |  |
| 1. NH46 | |  | 1. NH53 | |

**图1 本实验使用的4种去雾实验图像数据，其中左侧位有雾图像（实验代码中标注为Hazy），右侧为无雾图像（实验代码中标注为Ground Truth或GT）**

其中，不同方法在Forest的实验结果已经作为范例给出，图像去雾结果及其亮度直方图如图2所示，客观质量评价指标如表1所示。

请注意，与以往对灰度图像的处理方式不同，本实验图像为RGB图像。在使用histeq()和imhist()函数前，请先使用rgb2hsi()函数将待处理的彩色图像从RGB空间转换到HSI空间，单独将亮度通道及“I”通道提取出来。在本实验中，只须对“I”通道图像进行去雾即可。去雾完成后，再将去雾后“I”通道图像与原来“H”通道和“S”通道进行拼接，最终形成完整的HSI去雾图像。

**现要求完成以下任务：**

**1**. 使用直方图均衡化、直方图规定化、以及暗通道先验算法完成在NH50、NH46、以及NH53三张图像的去雾实验，并展示去雾图像及其亮度直方图，排版格式应与图2保持一致；同时，计算每种方法的客观图像质量评价指标RMSE、PSNR、SSIM、记录其运行时间Time，并补全表1。

**2**. 结合你生成的去雾图像、亮度直方图、和对应的客观评价质量指标，分析每种方法的优劣。

**3**. 思考并回答问题：图像的客观质量评价指标好，其主观视觉效果就一定好吗？客观评价质量指标是否能完全反映出一个算法的图像去雾结果的优劣？结合本实验结果阐明原因。

**4.** 思考并回答问题：在本实验中，直方图规定化方法通过将有雾原图与无雾原图进行匹配，生成了去雾图像，以图2第四列所示图像为例。将有雾原图与无雾原图进行匹配从而达到去雾效果，你认为这种方法是否具有实际应用价值？

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 墙上挂着一幅画  中度可信度描述已自动生成 | | | | |
| 图表, 条形图, 直方图  描述已自动生成 | | | | |
| (a) | (b) | (c) | (d) | (e) |

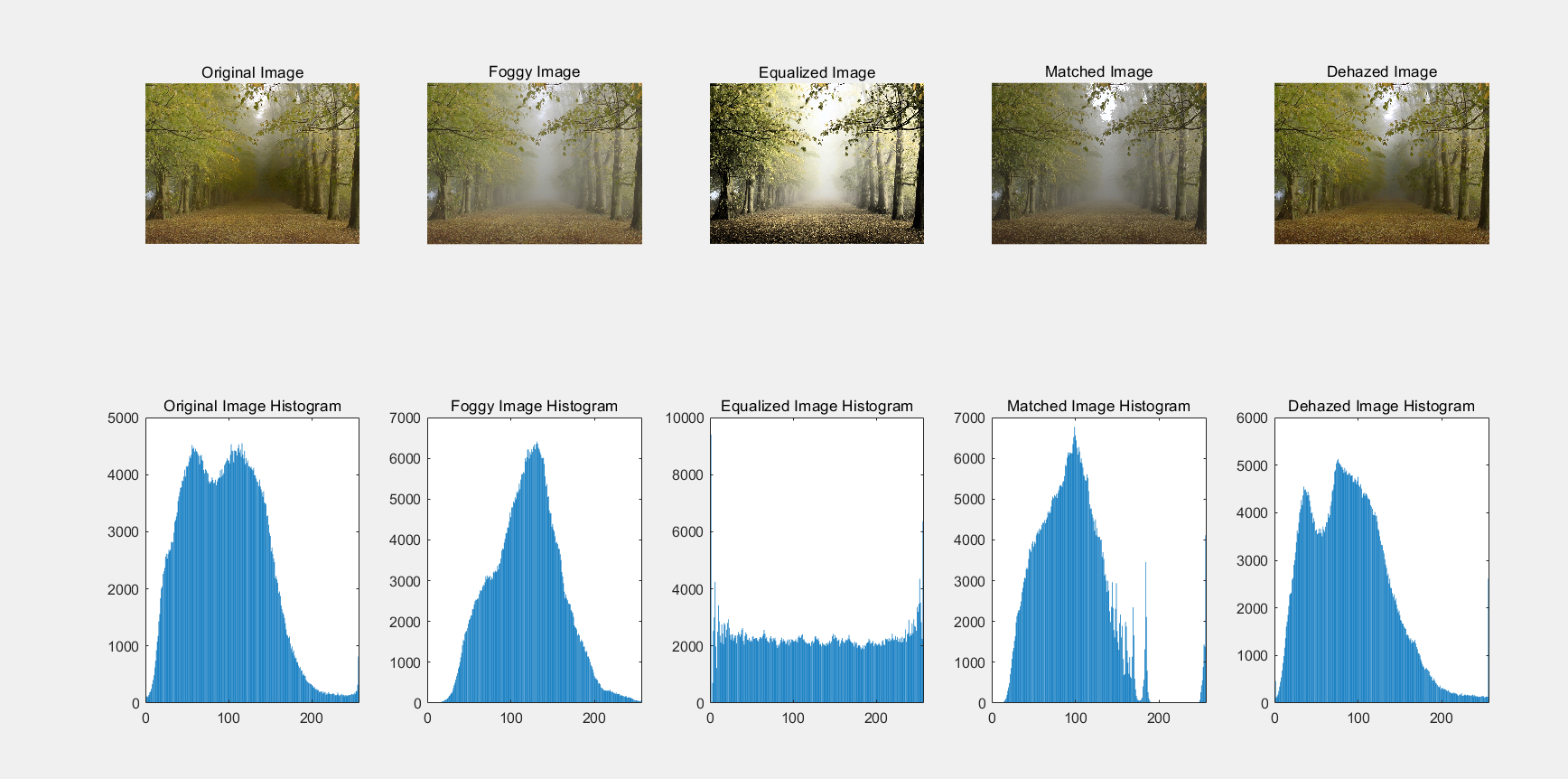
**图2 （示例）各种方法在Forest上的去雾结果及其直方图。(a)有雾原图；(b)无雾原图；(c)直方图均衡化；(d)直方图规定化；(e)暗通道先验算法。**

## 三、实验分析

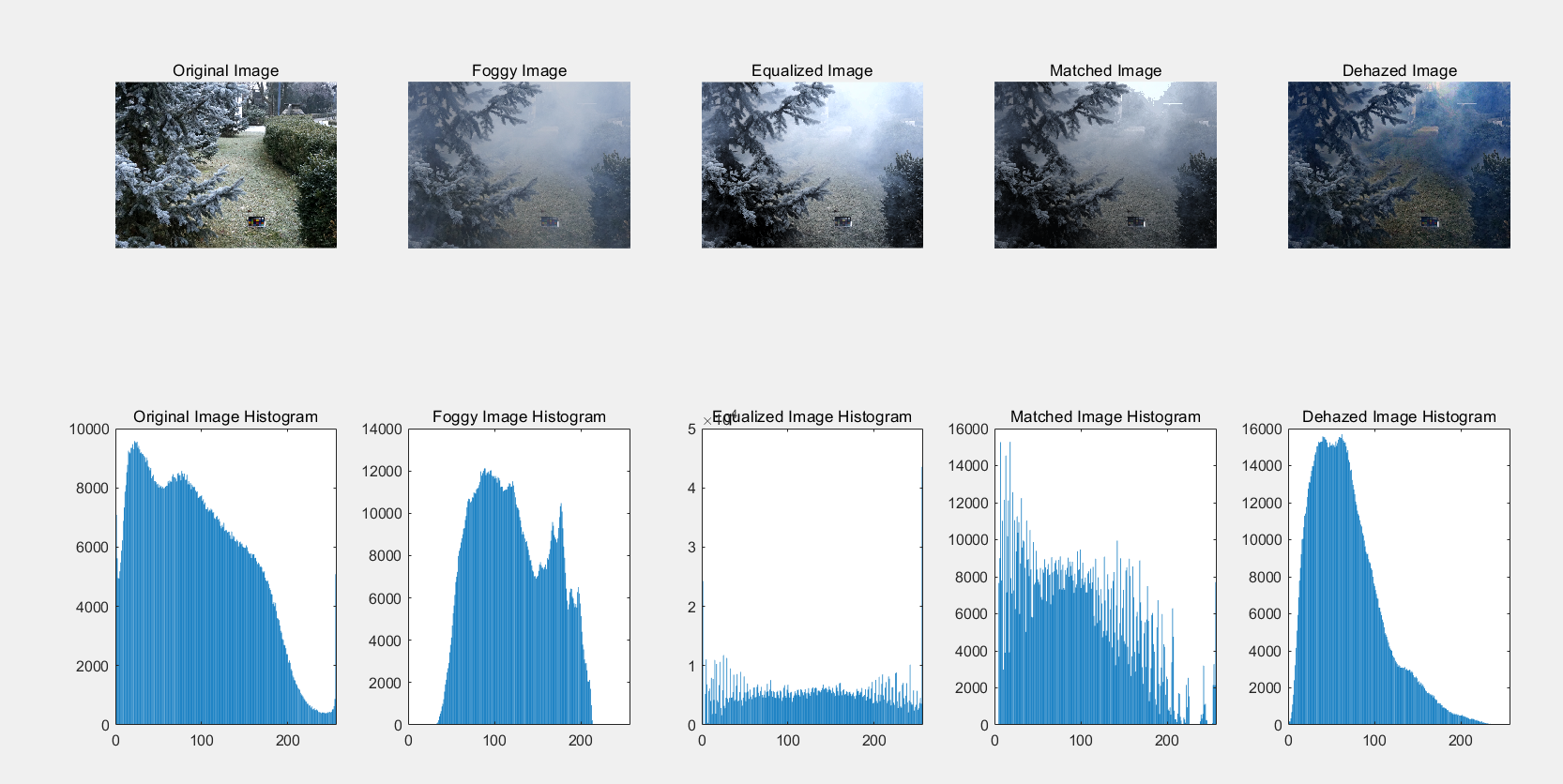
根据以上实验任务，可以采取imread、rgb2hsi等函数读取和处理图像，并利用histeq函数进行直方图均衡和直方图规定化，利用所给定的代码中的暗通道先验算法进行去雾，显示和比较这几种方法的结果。

## 四、实验结果

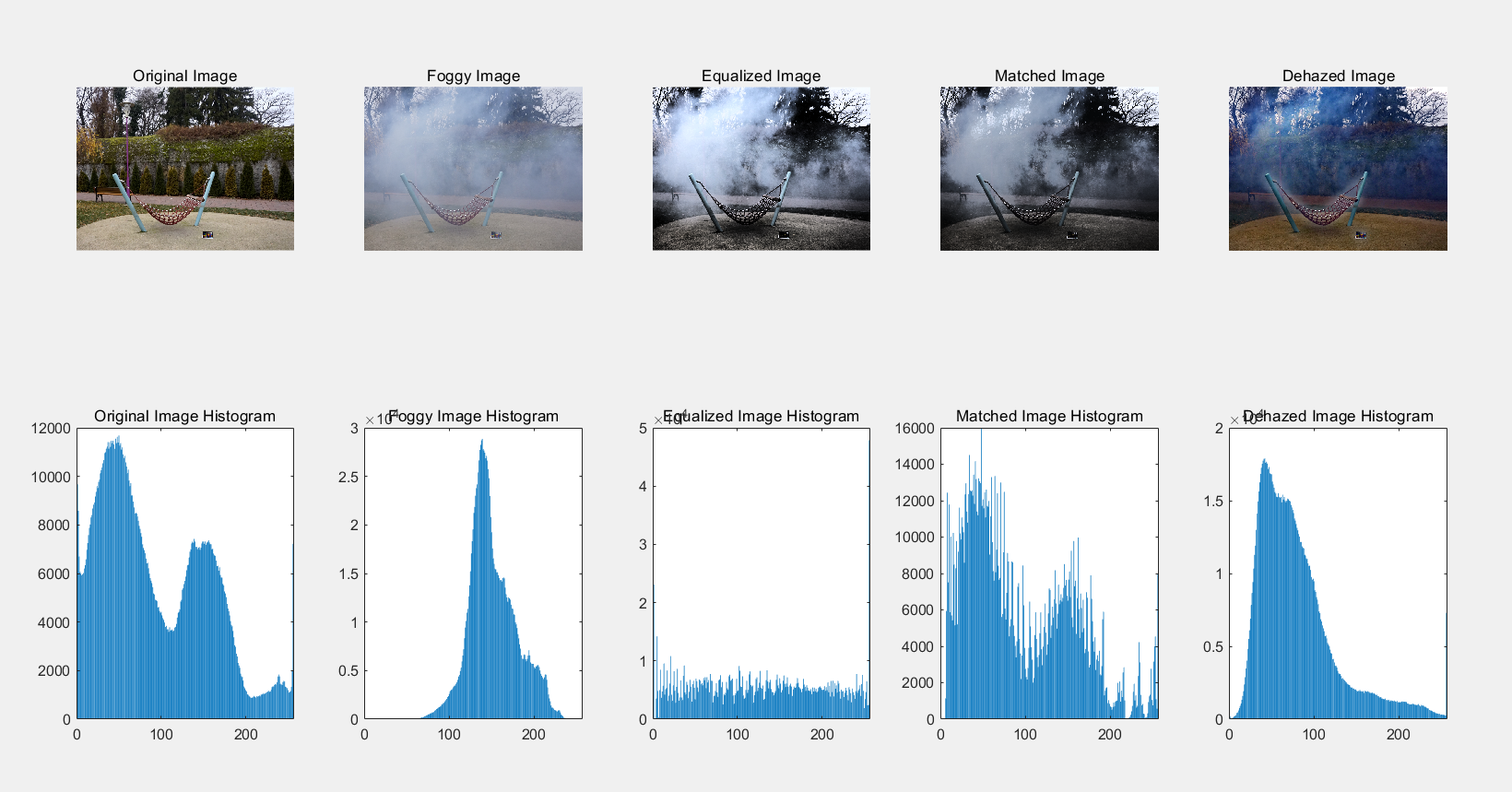
Forest(示例)实验结果：



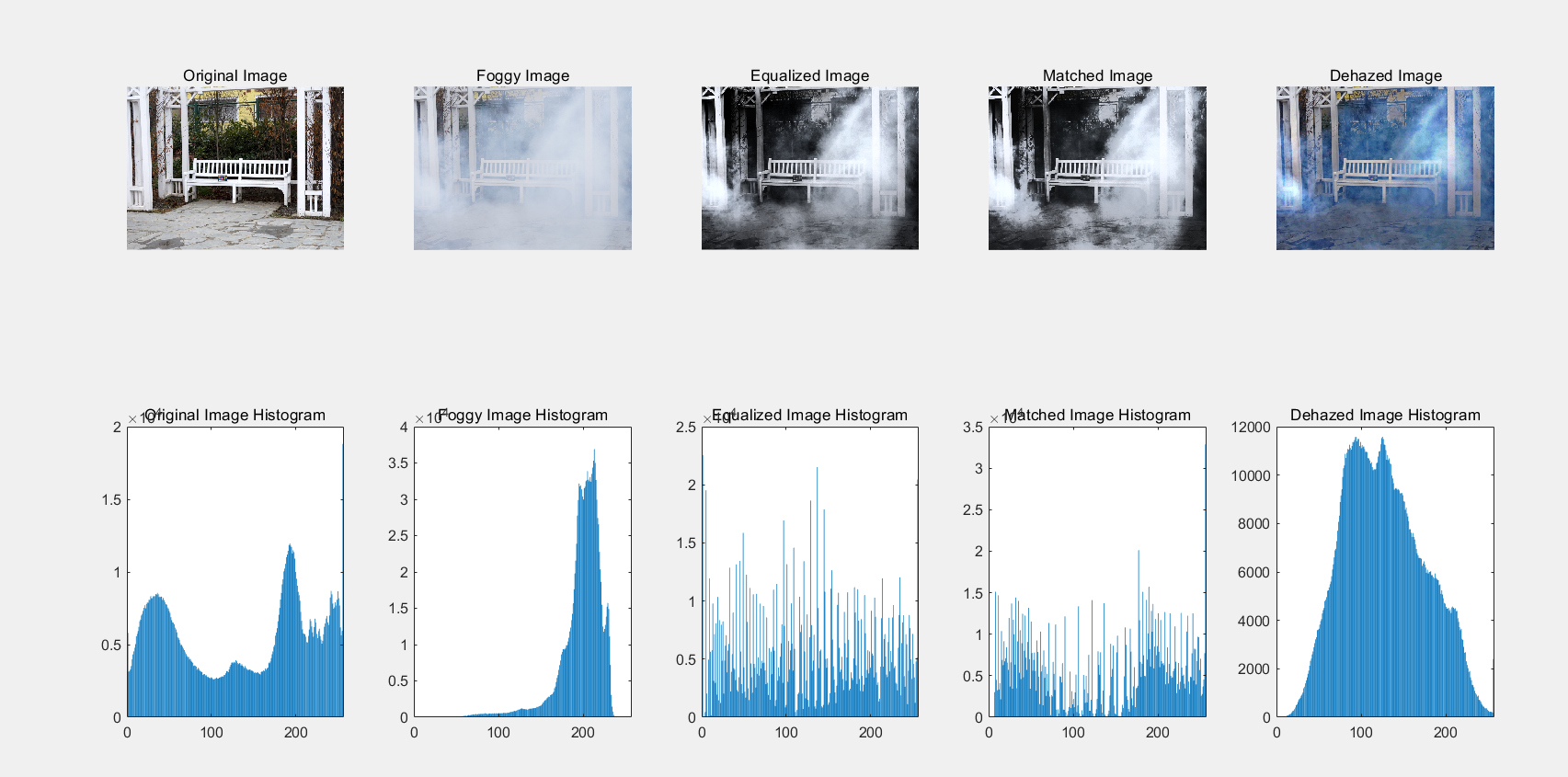
NH50实验结果：



NH53实验结果：



NH46实验结果:



**表1 各种方法在不同图像数据上的客观质量评价指标结果**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Data** | **Metric** | **Original** | **HistEq** | **HistSp** | **Dark Channel** | **(Fast) Dark Channel** |
| Forest  (Rescaled: 0.5) | RMSE↓ | 35.888 | 67.768 | 27.413 | 18.727 | **14.099** |
| PSNR↑ | 16.684 | 11.368 | 18.368 | 22.319 | **24.605** |
| SSIM↑ | 0.834 | 0.671 | 0.779 | 0.933 | **0.954** |
| Time (s)↓ | - | **0.045** | 0.047 | 23.991 | 4.123 |
| NH50  (Rescaled: 0.5) | RMSE↓ | 58.476 | 74.219 | 58.291 | 55.530 | **55.066** |
| PSNR↑ | 12.660 | 10.634 | 12.727 | 12.962 | **13.018** |
| SSIM↑ | 0.381 | **0.488** | 0.436 | 0.357 | 0.419 |
| Time (s)↓ | - | **0.134** | 0.196 | 72.027 | 10.064 |
| NH46  (Rescaled: 0.5) | RMSE↓ | 99.077 | 89.461 | 92.373 | 74.139 | **73.112** |
| PSNR↑ | 8.118 | 9.029 | 8.752 | 10.118 | **10.203** |
| SSIM↑ | 0.279 | **0.371** | 0.358 | 0.225 | 0.251 |
| Time (s)↓ | - | 0.167 | **0.139** | 65.221 | 3.554 |
| NH53  (Rescaled: 0.5) | RMSE↓ | 83.706 | 96.580 | 77.080 | 54.799 | **54.097** |
| PSNR↑ | 9.404 | 8.240 | 10.161 | 12.456 | **12.525** |
| SSIM↑ | 0.182 | 0.227 | **0.233** | 0.191 | 0.216 |
| Time (s)↓ |  | 0.144 | **0.130** | 37.729 | 3.099 |

**注：每个数据集括弧中的数字表示图像放缩尺度，如(Rescaled: 0.5)表示将原图像的长和宽放缩到0.5倍。**

如表所示，暗通道先验算法基本拥有最佳的客观质量评价指标。在实践中，暗通道先验算法（快速）拥有更快的速度和更佳的评价指标。但是，算法的优劣还需要结合主观视觉效果进行进一步判断，总体上暗通道先验算法具有非常不错的效果和性能。

## 五、实验思考题

**图像的客观质量评价指标好，其主观视觉效果就一定好吗？客观评价质量指标是否能完全反映出一个算法的图像去雾结果的优劣？结合本实验结果阐明原因。**

客观质量评价指标是一种用于量化图像质量的方法，通常基于图像的像素值和结构等特征。这些指标包括但不限于 RMSE（均方根误差）、PSNR（峰值信噪比）、SSIM（结构相似性指标）等。然而，客观评价指标并不能完全反映出图像的主观视觉效果，也不能充分代表人类对图像质量的感知。这是因为人类视觉系统对图像质量的感知受到许多因素的影响，如颜色、对比度、清晰度、细节等，而客观评价指标无法全面反映出人类对图像的整体感知。其次，客观评价指标通常是基于一般性的图像质量标准而设计的，但不同的应用场景可能对图像质量有不同的要求。再者，不同的观察者可能会因为个人喜好、背景知识等因素对同一张图像有不同的评价。

在本实验中，客观评价指标可以帮助我们量化不同去雾算法的性能，但并不能完全代替人类的主观评价。例如，某种算法在客观评价指标上表现良好，但在视觉上可能出现了一些异常或者失真，导致人类观察者不满意（如HistEq在NH50、NH46图像中的SSIM指标表现较好，但实际观感不佳。在Forest图像中，HistEq的表现均非最佳，但其观感可能优于Dark Channel算法）。因此，综合考虑客观评价指标和主观视觉感知，可以更全面地评估图像去雾算法的优劣。

**在本实验中，直方图规定化方法通过将有雾原图与无雾原图进行匹配，生成了去雾图像，以图2第四列所示图像为例。将有雾原图与无雾原图进行匹配从而达到去雾效果，你认为这种方法是否具有实际应用价值？**

直方图规定化方法具有一定的实际应用价值。然而，这种方法要求有无雾原图作为参考，但在实际应用场景中获取无雾原图可能不太容易。方法的效果可能会受到场景和条件的影响，因此需要针对具体场景进行调整和优化。

在实际应用中，这种方法更容易被用于以下场景：

对于某日拍摄的有雾图像A，需要还原其中的某些细节。在无雾时拍摄相同场景相同位置的图像B，以图像B为参考对A进行直方图规定化，以试图还原其中的细节。

## 六、实验总结

在本实验中，通过直方图均衡、直方图规定化和暗通道先验算法分别尝试了图像去雾，并通过客观评价指标比较了它们的优劣。实验结果显示，暗通道先验算法拥有最佳的客观评价指标和不错的视觉效果，是图像去雾的极佳选择之一。

## 附：实验代码

clear;

clc;

% 读取含雾图和原图

hazy\_img = double(imread('forest.jpg'))/255;

GT\_img = double(imread('forest\_recovered.jpg'))/255;

hazy\_img = imresize(hazy\_img, 0.5);

GT\_img = imresize(GT\_img, 0.5);

% 直方图均衡化

tic;

hazy\_img\_hsi = rgb2hsi(hazy\_img);

hazy\_img\_hsi(:,:,3) = histeq(hazy\_img\_hsi(:,:,3));

hazy\_img\_equalized = hsi2rgb(hazy\_img\_hsi);

% 计算均衡化后图像的评价指标

equalized\_time = toc;

equalized\_rmse = rmse(GT\_img, hazy\_img\_equalized)\*255;

equalized\_psnr = psnr(GT\_img, hazy\_img\_equalized);

equalized\_ssim = ssim(GT\_img, hazy\_img\_equalized);

% 直方图规定化

tic;

GT\_img\_hsi = rgb2hsi(GT\_img);

hazy\_img\_hsi(:,:,3) = histeq(hazy\_img\_hsi(:,:,3), imhist(GT\_img\_hsi(:,:,3)));

hazy\_img\_matched = hsi2rgb(hazy\_img\_hsi);

% 计算规定化后图像的评价指标

matched\_time = toc;

matched\_rmse = rmse(GT\_img, hazy\_img\_matched)\*255;

matched\_psnr = psnr(GT\_img, hazy\_img\_matched);

matched\_ssim = ssim(GT\_img, hazy\_img\_matched);

% 暗通道先验算法

tic;

dehazed\_img = dehaze(hazy\_img);

% 计算暗通道先验算法去雾后图像的评价指标

dehazed\_time = toc;

dehazed\_rmse = rmse(GT\_img, dehazed\_img)\*255;

dehazed\_psnr = psnr(GT\_img, dehazed\_img);

dehazed\_ssim = ssim(GT\_img, dehazed\_img);

% 暗通道先验算法（快速）

tic;

dehazed\_fast\_img = dehaze\_fast(hazy\_img);

% 计算暗通道先验算法（快速）去雾后图像的评价指标

dehazed\_fast\_time = toc;

dehazed\_fast\_rmse = rmse(GT\_img, dehazed\_fast\_img)\*255;

dehazed\_fast\_psnr = psnr(GT\_img, dehazed\_fast\_img);

dehazed\_fast\_ssim = ssim(GT\_img, dehazed\_fast\_img);

% 计算原图的评价指标

GT\_rmse = rmse(GT\_img, hazy\_img)\*255;

GT\_psnr = psnr(GT\_img, hazy\_img);

GT\_ssim = ssim(GT\_img, hazy\_img);

% 显示图像和直方图

figure;

% 显示原图

subplot(2,5,1);

imshow(GT\_img);

title('Original Image');

subplot(2,5,6);

bar(imhist(GT\_img));

title('Original Image Histogram');

% 显示含雾图像

subplot(2,5,2);

imshow(hazy\_img);

title('Foggy Image');

subplot(2,5,7);

bar(imhist(hazy\_img));

title('Foggy Image Histogram');

% 显示直方图均衡化后的图像

subplot(2,5,3);

imshow(hazy\_img\_equalized);

title('Equalized Image');

subplot(2,5,8);

bar(imhist(hazy\_img\_equalized));

title('Equalized Image Histogram');

% 显示直方图规定化后的图像

subplot(2,5,4);

imshow(hazy\_img\_matched);

title('Matched Image');

subplot(2,5,9);

bar(imhist(hazy\_img\_matched));

title('Matched Image Histogram');

% 显示暗通道先验算法去雾后的图像

subplot(2,5,5);

imshow(dehazed\_fast\_img);

title('Dehazed Image');

subplot(2,5,10);

bar(imhist(dehazed\_fast\_img));

title('Dehazed Image Histogram');

% 显示结果

disp('Original:');

disp(['rmse: ', num2str(GT\_rmse)]);

disp(['psnr: ', num2str(GT\_psnr)]);

disp(['ssim: ', num2str(GT\_ssim)]);

disp('Direct Histogram Equalization:');

disp(['rmse: ', num2str(equalized\_rmse)]);

disp(['psnr: ', num2str(equalized\_psnr)]);

disp(['ssim: ', num2str(equalized\_ssim)]);

disp(['Time: ', num2str(equalized\_time), ' seconds']);

disp('Histogram Matching:');

disp(['rmse: ', num2str(matched\_rmse)]);

disp(['psnr: ', num2str(matched\_psnr)]);

disp(['ssim: ', num2str(matched\_ssim)]);

disp(['Time: ', num2str(matched\_time), ' seconds']);

disp('Dark Channel Prior:');

disp(['rmse: ', num2str(dehazed\_rmse)]);

disp(['psnr: ', num2str(dehazed\_psnr)]);

disp(['ssim: ', num2str(dehazed\_ssim)]);

disp(['Time: ', num2str(dehazed\_time), ' seconds']);

disp('Dark Channel Prior(Fast):');

disp(['rmse: ', num2str(dehazed\_fast\_rmse)]);

disp(['psnr: ', num2str(dehazed\_fast\_psnr)]);

disp(['ssim: ', num2str(dehazed\_fast\_ssim)]);

disp(['Time: ', num2str(dehazed\_fast\_time), ' seconds']);