

# Enhancing Images

The image enhancement techniques in this reading are focused more on making your images more clear for human observers. These techniques are not intended to directly improve image segmentation. Image processing is often the first step in a larger *computer vision* project. Part of computer vision is replicating and automating the tasks of human vision. In this context, increasing the visual clarity of an image may be an important preprocessing step.

Note: The techniques in this reading are more complex than those previously shown. The details of the algorithms involved are beyond the scope of this course.

## Table of Contents

Haze Reduction.....	1
Brightening Dark Images.....	2

## Haze Reduction

Atmospheric haze can obscure the details in an image. The `imreducehaze` function reduces the effects of fog, smoke, and haze.

```
foggy = imread("foggyroad.jpg");
defogged = imreducehaze(foggy);
montage({foggy, defogged})
```



This photo of Back Bay in Boston was taken on a hazy winter morning.

```
haze = imread("boston_haze.jpg");
dehaze = imreducehaze(haze);
montage({haze, dehaze})
```



Snow and frost on objects in an image can have similar effects to atmospheric contaminants. The `imreducehaze` function can also improve the visibility in these cases.

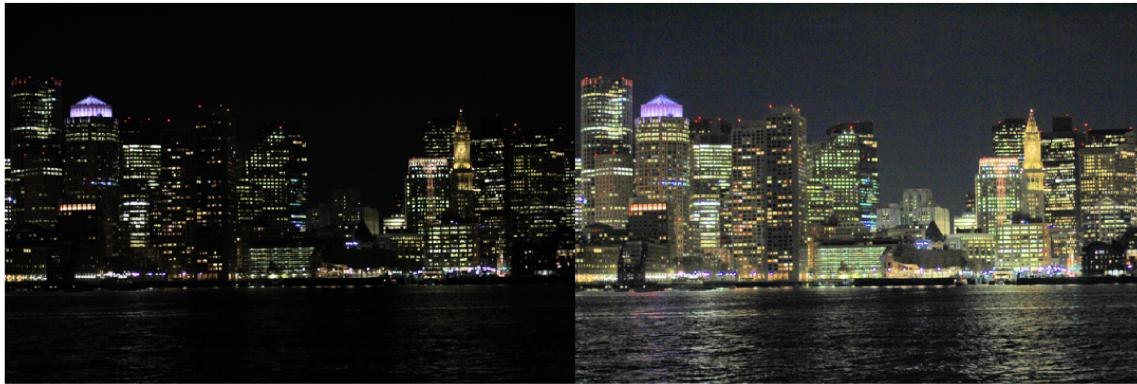
```
snowy1 = imread("mountain1.jpg");
cleared1 = imreducehaze(snowy1);
montage({snowy1, cleared1})
```



## Brightening Dark Images

Underexposed images use only part of the available brightness range. The `imadjust` function can increase the clarity of these images. But some dark images, such as the Boston waterfront image, are not underexposed. For images like these, the `imlocalbrighten` function can increase the apparent lighting of the image to reveal more of the subjects. The function takes an optional parameter, `amount`, that determines how much the brightness is increased. This variable takes values between 0 and 1, with 1 being the default.

```
darkCity = imread("boston_night.jpg");
amount = 0.9;
brightCity = imlocalbrighten(darkCity, amount);
montage({darkCity, brightCity})
```



The lighting conditions when an image is taken can sometimes be out of your control. For example, collecting nocturnal wildlife images using a flash could negatively impact the data. Or your otherwise cheerful dog could be sitting under a bush in the dark being creepy for no good reason.

```
darkDog = imrotate(imread("dash.jpg"), -90);
```

Note that `imread` does not look at the metadata attached to the file to determine the image orientation. Images from cell phones and other devices may need to be rotated to be viewed correctly.

```
imshow(darkDog)
```



For images like these, the `imlocalbrighten` function can artificially increase the lighting in the scene.

```
brightDog = imlocalbrighten(darkDog);  
montage({darkDog, brightDog})
```



The `imreducehaze` and `imlocalbrighten` functions are not meant to replace the contrast adjustment technique presented earlier. They are computationally intensive and can add significant noise to an image.

*Copyright 2022 The MathWorks, Inc.*