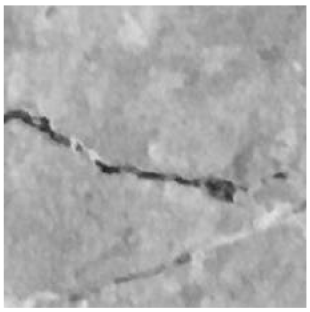
Import the image and convert it to grayscale. Segment the image using the *imbinarize* function with the default settings. The lighter background will be selected as the foreground. Invert the image so that the crack is selected as the foreground. This mask will be the baseline for this set of questions.

How many pixels are in the crack?

```
img = imread("00035.jpg");
img = im2gray(img);
imshow(img);
```



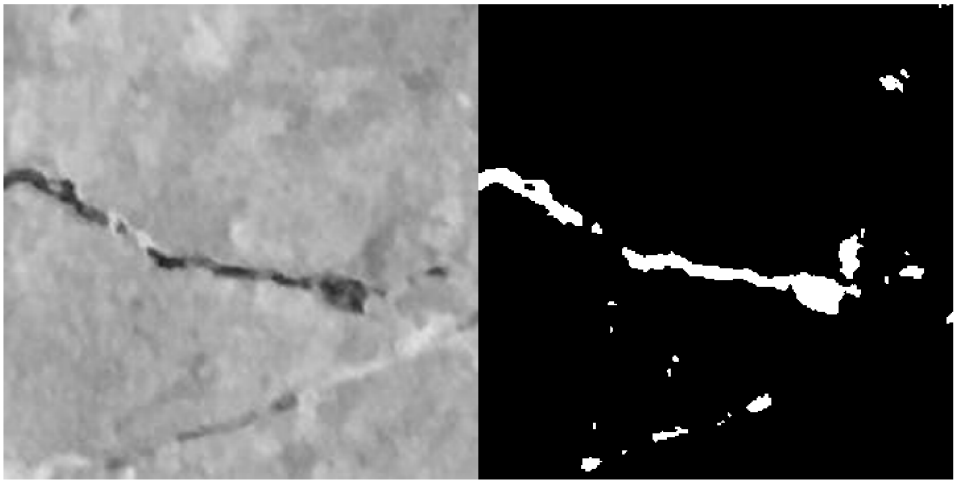
```
BW = imbinarize(img)
```

```
BW = 227×227 logical array
   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1 ···
   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1
     ⋮
```

```
img1 = ~BW
```

```
img1 = 227×227 logical array
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0 ···
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
     ⋮
```

```
montage({img,img1})
```



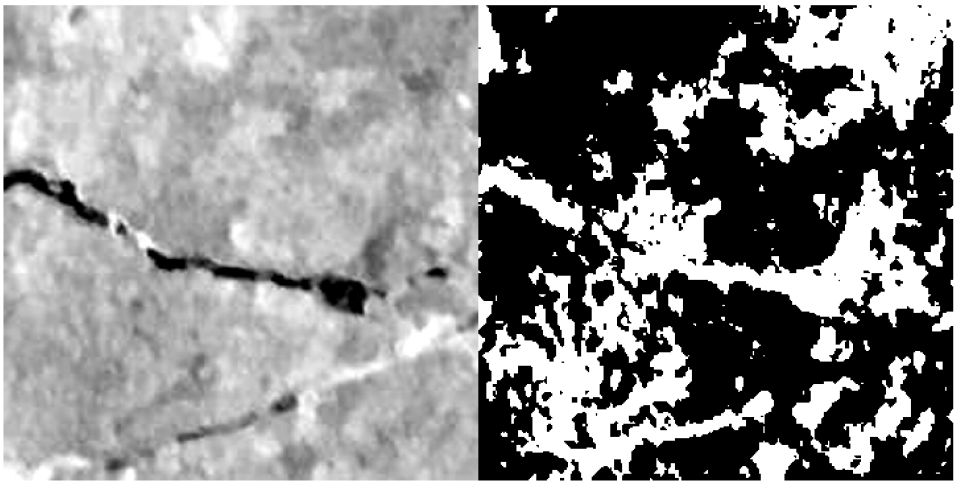
```
nnz(img1)
```

```
ans = 1800
```

Adjust the contrast of the grayscale image *00035.jpg* by stretching the histogram. Then perform the same segmentation and inversion as in Question 1.

How many pixels are in the crack in this segmentation?

```
img = imread("00035.jpg");
img = im2gray(img);
imgAdj = imadjust(img);
BW = imbinarize(imgAdj);
img1 = ~BW;
montage({imgAdj,img1});
```



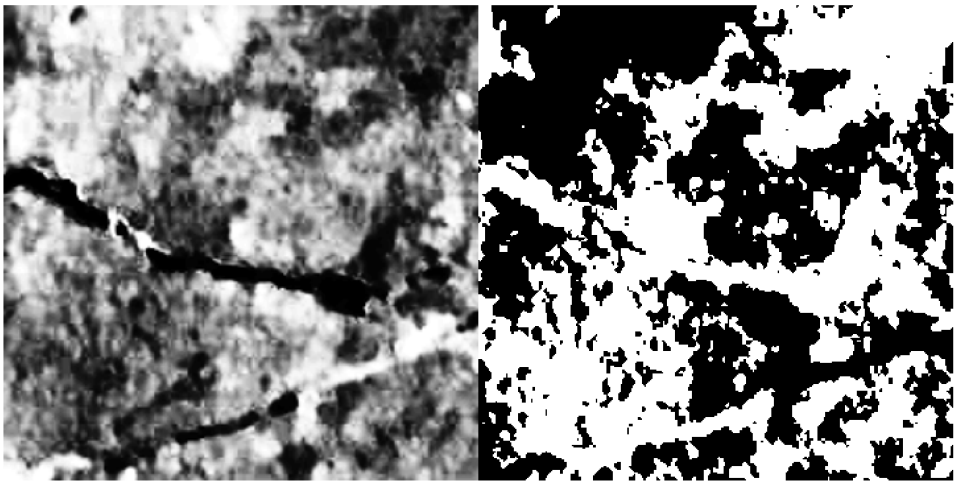
```
nnz(img1);
```

Question 3

Equalize the image histogram of the grayscale image *00035.jpg*. Perform the same segmentation and inversion as in Question 1.

How many pixels are in the crack in this segmentation?

```
img = imread("00035.jpg");
img = im2gray(img);
imgEq = histeq(img);
BW = imbinarize(imgEq);
img1 = ~BW;
montage({imgEq,img1});
```



```
nnz(img1)
```

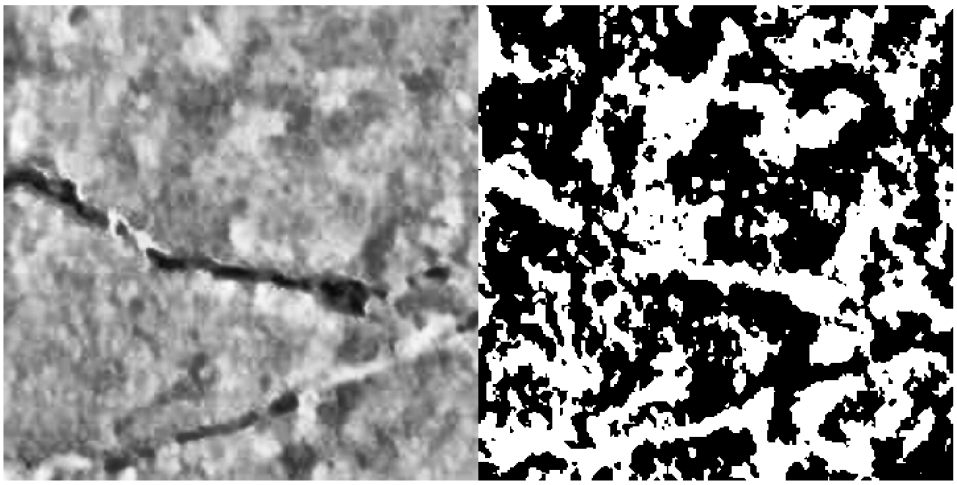
```
ans = 25780
```

Question 4

Perform adaptive histogram equalization on the grayscale image *00035.jpg*. Perform the same segmentation and inversion as in Question 1.

How many pixels are in the crack in this segmentation?

```
img = imread("00035.jpg");
img = im2gray(img);
imgah = adapthisteq(img);
BW = imbinarize(imgah);
img1 = ~BW;
montage({imgah,img1});
```



```
nnz(img1)
```

```
ans = 22638
```

Use the *imlocalbrighten* function to increase the brightness of the *car_3.jpg* image. Calculate the difference in the average luminance of the image before and after the transformation.

Recall: to find the average luminance, find the luminance value of each pixel by converting the image to grayscale, then average the resulting values.

```
car = imread("car_3.jpg");
car_brt = imlocalbrighten(car);
montage({car, car_brt})
```

```
car_gs = im2gray(car);
car_brt_gs = im2gray(car_brt)
```

```
car_brt_gs = 480×640 uint8 matrix
```

```
    11    11    11    11    11    11    11    11    11    11    11    11    11    11    11    11 ···
    11    11    11    11    11    11    11    11    11    11    11    11    11    11    11    11
    11    11    11    11    11    11    11    11    11    11    11    11    11    11    11    11
    11    11    11    11    11    11    11    11    11    11    11    11    11    11    11    11
    11    11    11    11    11    11    11    11    11    11    11    11    11    11    11    11
    11    11    11    11    11    11    11    11    11    11    11    11    11    11    11    11
    11    11    11    11    11    11    11    11    11    11    11    11    11    11    11    11
    11    11    11    11    11    11    11    11    11    11    11    11    11    11    11    11
    11    11    11    11    11    11    11    11    11    11    11    11    11    11    11    11
    11    11    11    11    11    11    11    11    11    11    11    11    11    11    11    11
      ⋮
```

```
car_avg_l = mean2(car_gs)
```

```
car_avg_l = 18.7302
```

```
car_brt_avg_l = mean2(car_brt_gs)
```

```
car_brt_avg_l = 74.2628
```

```
diff = car_brt_avg_l - car_avg_l
```

```
diff = 55.5325
```

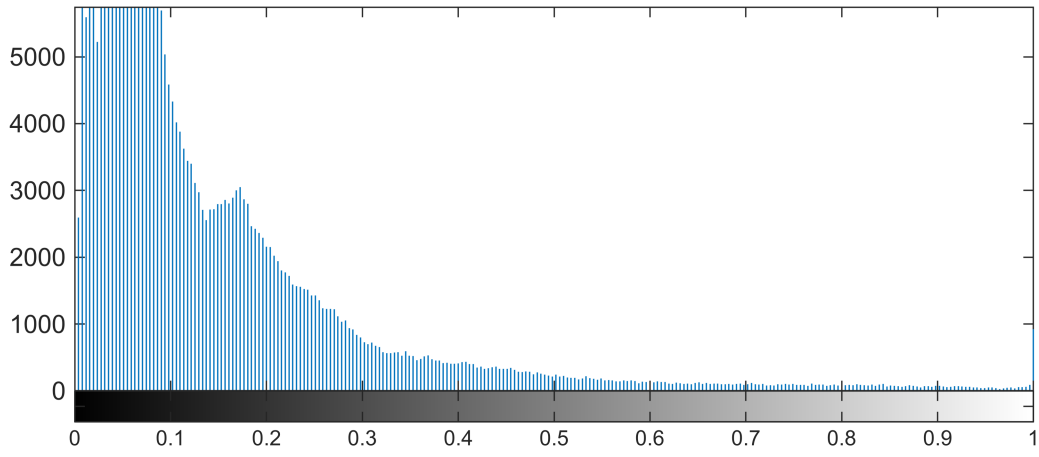
Load the car_2.jpg image and convert it to the HSV color space. Perform histogram equalization on the Value color plane. Calculate the difference between the average luminance of the image before and after the transformation.

Recall: Converting an image from RGB to HSV changes the data type from uint8 to double. You will need to convert the equalized image back to RGB AND convert the data type back to uint8.

```
car = imread("car_2.jpg");
car2 = im2gray(car);
car2_avg_lum = mean2(car2)
```

```
car2_avg_lum = 34.7940
```

```
car_hsv = rgb2hsv(car);
imhist(car_hsv(:,:,3));
```



```
car_back = hsv2rgb(car_hsv);
car_back = uint8(car_back);
car_back = im2gray(car_back)
```

```
car_back = 480×640 uint8 matrix
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0···
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
     ⋮
```

```
car_new_avg_lum = mean2(car_back)
```

```
car_new_avg_lum = 0.0296
```

```
differ = (car2_avg_lum - car_new_avg_lum)
```

```
differ = 34.7644
```

Use the *imreducehaze* function to remove some of the snow and frost from the *mountain2.jpg* image. Calculate the difference between the average luminance of the image before and after the transformation.

Note: In this case, we are darkening the image, so the average luminance will be lower in the transformed image. Enter a positive value for the difference in the average luminance values.

```
m = imread("mountain2.jpg")
```

```
m = 1932×2576×3 uint8 array
m(:,:,1) =

  Columns 1 through 1,666

   223   223   223   224   224   225   225   225   225   225   225   225   225   225   224   224   224   224   224
     ⋮
```

```
m = im2gray(m);
m_avg = mean2(m)
```

```
m_avg = 155.8008
```

```
new_m = imreducehaze(m);
new_m = im2gray(new_m);
new_m_avg = mean2(new_m)
```

```
new_m_avg = 105.7670
```

```
dif = (new_m_avg - m_avg)
```

```
dif = -50.0338
```

How many guitar picks are present in *picks.jpg*?

```
guitar = imread("picks.jpg");
guitar = im2gray(guitar);
guitar_adj = imadjust(guitar);
imshow(guitar_adj)
```