

Contents

- Read Image and Display Seperate RGB Components
- Trahanias Method set up
- Histogram -- frequency of intensity levels -- RED
- Histogram -- frequency of intensity levels -- GREEN
- Histogram -- frequency of intensity levels -- BLUE
- final image compared with histeq()

```
% determine sample image locations
% folder = fileparts(which('cameraman'));
clc
clear
```

Read Image and Display Seperate RGB Components

```
I = imread('onion.png');

R = I(:, :, 1);
G = I(:, :, 2);
B = I(:, :, 3);

rows      = size(I, 1);
columns   = size(I, 2);
a         = uint8(zeros( rows, columns ));
red_final = uint8(zeros( rows, columns ));
green_final = uint8(zeros( rows, columns ));
blue_final = uint8(zeros( rows, columns ));

L         = 256; % number of intensity levels
a_n       = rows*columns; % total number of pixels

red_comp  = cat(3, R, a, a);
green_comp = cat(3, a, G, a);
blue_comp = cat(3, a, a, B);

figure
subplot(2,2,1)
imshow(I), title('Original Image')

subplot(2,2,2)
imshow(red_comp), title('Red Component');

subplot(2,2,3)
imshow(green_comp), title('Green Component');

subplot(2,2,4)
imshow(blue_comp), title('Blue Component');
```

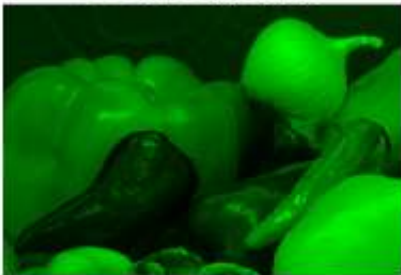
Original Image



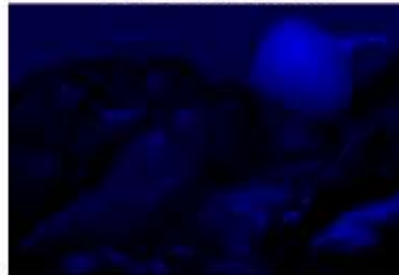
Red Component



Green Component



Blue Component



Trahanias Method set up

Histogram -- frequency of intensity levels -- RED

```
freq_r = zeros(L, 1);
pdf_r  = zeros(L, 1);
cdf_r  = zeros(L, 1);
cum_r  = zeros(L, 1);
out_r  = zeros(L, 1);

for i = 1:rows
    for j = 1:columns
        value = R(i, j);
        freq_r(value + 1) = freq_r(value + 1) + 1;
        pdf_r(value + 1) = freq_r(value + 1) / a_n;
    end
end

sum = 0;
for i = 1:size(pdf_r)
    sum = sum + freq_r(i);
    cum_r(i) = sum;
    cdf_r(i) = cum_r(i) / a_n;
    out_r(i) = round(cdf_r(i) * (L - 1));
end

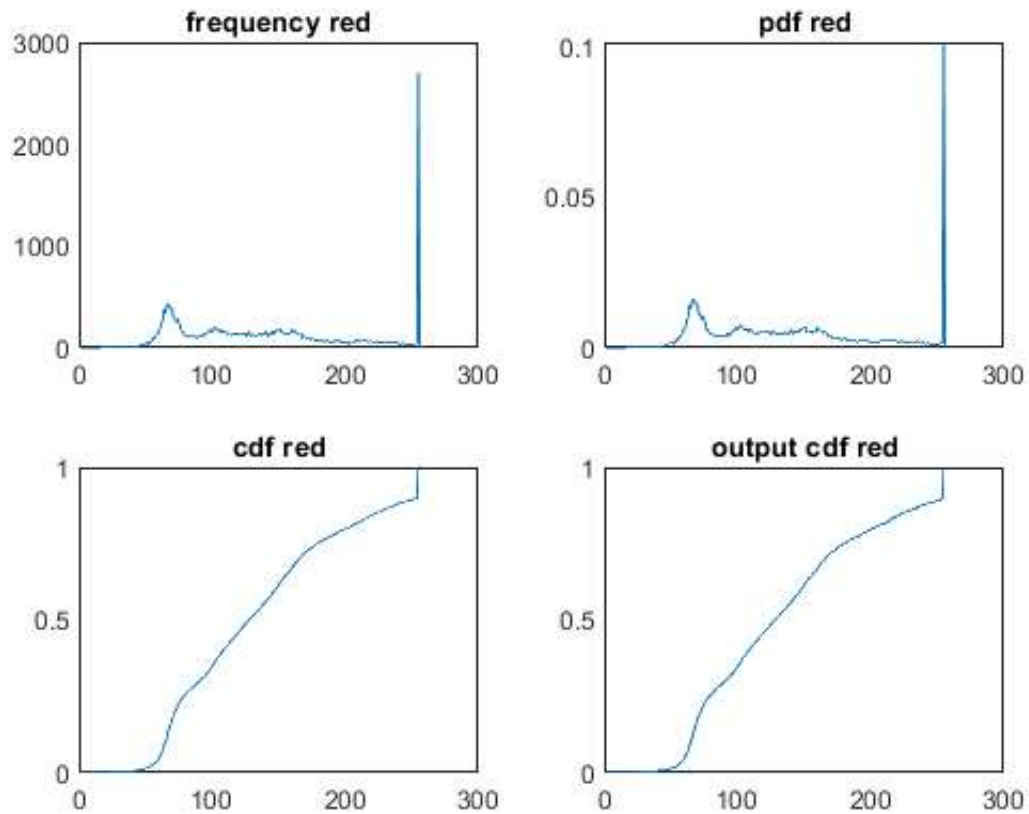
for i = 1:rows
    for j = 1:columns
```

```

        red_final(i, j) = out_r(R(i, j) + 1);
    end
end

figure
subplot(2,2,1)
plot(freq_r), title('frequency red')
subplot(2,2,2)
plot(pdf_r), title('pdf red')
subplot(2,2,3)
plot(cdf_r), title('cdf red')
subplot(2,2,4)
plot(out_r/256), title('output cdf red')

```



Histogram -- frequency of intensity levels -- GREEN

```

freq_g = zeros(L, 1);
pdf_g = zeros(L, 1);
cdf_g = zeros(L, 1);
cum_g = zeros(L, 1);
out_g = zeros(L, 1);

for i = 1:rows
    for j = 1:columns
        value = G(i, j);
        freq_g(value + 1) = freq_g(value + 1) + 1;
        pdf_g(value + 1) = freq_g(value + 1) / a_n;
    end
end

```

```

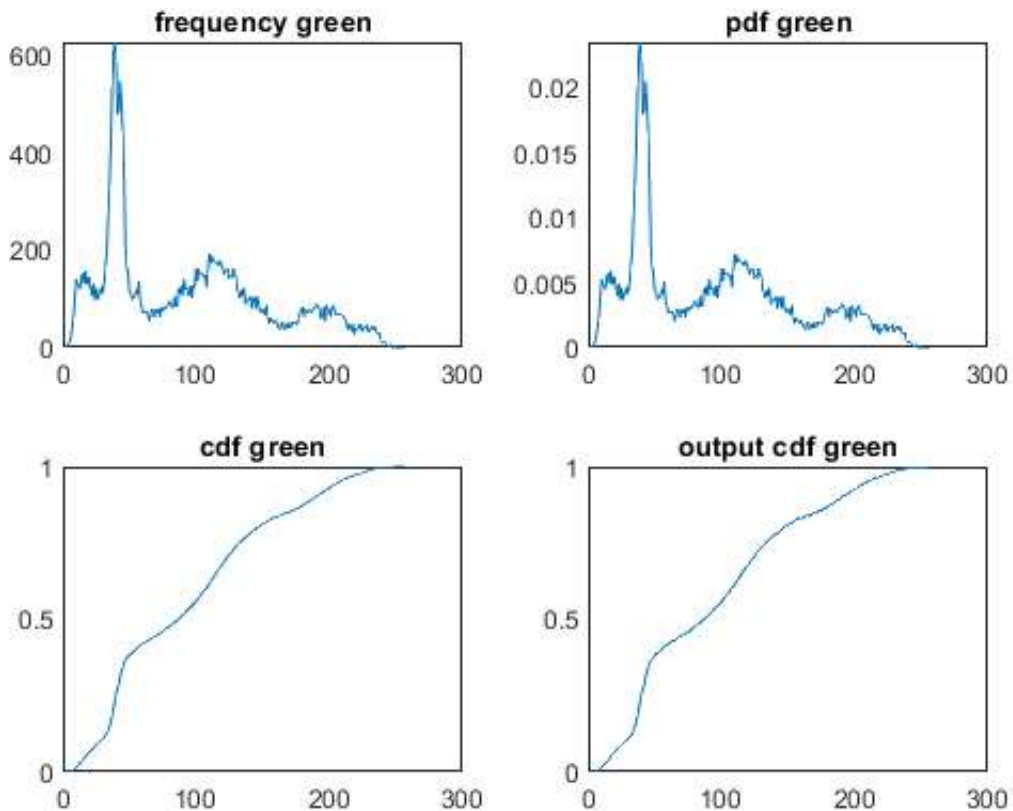
end
end

sum = 0;
for i = 1:size(pdf_g)
    sum      = sum + freq_g(i);
    cum_g(i) = sum;
    cdf_g(i) = cum_g(i) / a_n;
    out_g(i) = round(cdf_g(i) * (L - 1));
end

for i = 1:rows
    for j = 1:columns
        green_final(i, j) = out_g(G(i, j) + 1);
    end
end

figure
subplot(2,2,1)
plot(freq_g), title('frequency green')
subplot(2,2,2)
plot(pdf_g), title('pdf green')
subplot(2,2,3)
plot(cdf_g), title('cdf green')
subplot(2,2,4)
plot(out_g/256), title('output cdf green')

```



```

freq_b = zeros(L, 1);
pdf_b  = zeros(L, 1);
cdf_b  = zeros(L, 1);
cum_b  = zeros(L, 1);
out_b  = zeros(L, 1);

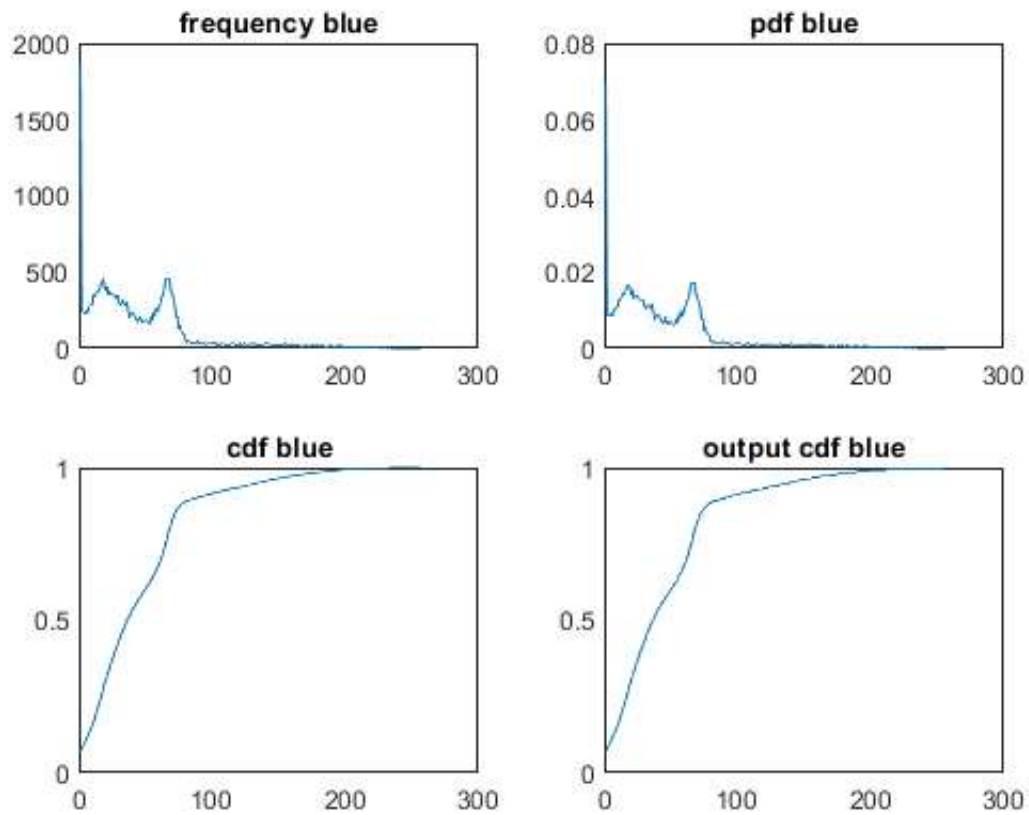
for i = 1:rows
    for j = 1:columns
        value      = B(i, j);
        freq_b(value + 1) = freq_b(value + 1) + 1;
        pdf_b(value + 1)  = freq_b(value + 1) / a_n;
    end
end

sum = 0;
for i = 1:size(pdf_b)
    sum      = sum + freq_b(i);
    cum_b(i) = sum;
    cdf_b(i) = cum_b(i) / a_n;
    out_b(i) = round(cdf_b(i) * (L - 1));
end

for i = 1:rows
    for j = 1:columns
        blue_final(i, j) = out_b(B(i, j) + 1);
    end
end

figure
subplot(2,2,1)
plot(freq_b), title('frequency blue')
subplot(2,2,2)
plot(pdf_b), title('pdf blue')
subplot(2,2,3)
plot(cdf_b), title('cdf blue')
subplot(2,2,4)
plot(out_b/256), title('output cdf blue')

```



final image compared with histeq()

```
final_image = cat(3, red_final, green_final, blue_final);
he = histeq(I);
figure
subplot(2,3,1)
imshow(I), title('original image')
subplot(2,3,2)
imshow(he), title('histeq function')
subplot(2,3,3)
imshow(final_image), title('trahs definition')

% results of final outputs
subplot(2,3,4)
imhist(I)
subplot(2,3,5)
imhist(he)
subplot(2,3,6)
imhist(final_image)
```

