

**MATLAB Code to Detect Different Object in an Image**

```

I = imread('final1.jpg');
I = I(:,:,1);
soi = size(I);           %size of image
IN = zeros(soi(1,1));   %initialize new matrix of size equal to image
R = zeros(2);          %initialize new matrix for equivalence table
s=1;                   %represents rows in equivalence table
t=1;                   %represents rows in equivalence table
m=0;                   %use to detect & count new object

%% Detection of Black Pixel in Image
for i = 1:soi(1,1);     %i represent rows in Image matrix, from 1 to size of
image
    for j = 1:soi(1,2); %j represent rows in Image matrix
        if I(i,j) < 20 ;    % Check pixel's colour (if Gray level less than
10)

%When black pixel is detected, now cheking its neighbor pixel
        if (IN(i-1,j-1)==0 && IN(i-1,j)==0 && IN(i-1,j+1)==0 && IN(i,j-
1)==0); %when all neighbor pixels are 0
            m=m+1;          %new object detected
            IN(i,j) = m; %replace the value of pixel in New Matrix on the
place where new object is detected
            end

            if (IN(i-1,j-1)== IN(i-1,j) && IN(i-1,j-1)==IN(i,j-1) && IN(i-
1,j-1)==IN(i-1,j+1) && IN(i-1,j)==IN(i-1,j+1) && IN(i-1,j)==IN(i,j-1) &&
IN(i-1,j+1)==IN(i,j-1)); %when neighbors are of same value
                IN(i,j)=m; %replace the value of pixel in New Matrix on the
place where new object is detected
                end

                    %When neighbor pixels are not of
same value
                if (IN(i,j-1) ~= IN(i-1,j-1)); %compairing pixel of previous
colom in same row & pixel of previous colomn in previous row
                    if(IN(i,j-1)~=0); %if the value of previous colomn
pixel in the same row is other than zero
                        IN(i,j) = IN(i,j-1); %then place the same value in
current detected pixel
                            %This check is performed to make
equivalence table
                                if IN(i-1,j-1)~=0; %if the value of previous colomn
pixel in the previous row is other than zero
                                    if IN(i-1,j-1) < IN(i,j-1); %Then compare the value
of both pixels
                                        R(s,t) = IN(i-1,j-1); %place the smaller value
in the 1st colomn of equivalence table
                                            R(s,t+1) = IN(i,j-1); %place the larger value
in the 1st colomn of equivalence table
                                                s=s+1; %increment in rows of
equivalance table
                                                    end
                                                        if IN(i-1,j-1) > IN(i,j-1); %same as in above if
condition,

```

```

        R(s,t) = IN(i,j-1);
        R(s,t+1) = IN(i-1,j-1);
        s=s+1;
    end
end
end
if IN(i-1,j-1) ~= 0;
    IN(i,j) = IN(i-1,j-1);

    if IN(i-1,j-1)~=0;
        if IN(i-1,j-1) < IN(i,j-1);
            R(s,t) = IN(i-1,j-1);
            R(s,t+1) = IN(i,j-1);
            s=s+1;
        end
        if IN(i-1,j-1) > IN(i,j-1);
            R(s,t) = IN(i,j-1);
            R(s,t+1) = IN(i-1,j-1);
            s=s+1;
        end
    end
end
end

if (IN(i,j-1) ~= IN(i-1,j));
    if(IN(i,j-1) ~= 0);
        IN(i,j)=IN(i,j-1);

        if (IN(i-1,j) ~= 0);
            if IN(i-1,j)<IN(i,j-1);
                R(s,t) = IN(i-1,j);
                R(s,t+1) = IN(i,j-1);
                s=s+1;
            end
            if IN(i-1,j)>IN(i,j-1);
                R(s,t) = IN(i,j-1);
                R(s,t+1) = IN(i-1,j-1);
                s=s+1;
            end
        end
    end
end
if IN(i-1,j)~=0;
    IN(i,j)=IN(i-1,j);

    if (IN(i-1,j) ~= 0);
        if IN(i-1,j)<IN(i,j-1);
            R(s,t) = IN(i-1,j);
            R(s,t+1) = IN(i,j-1);
            s=s+1;
        end
        if IN(i-1,j)<IN(i,j-1);
            R(s,t) = IN(i,j-1);
            R(s,t+1) = IN(i-1,j-1);
            s=s+1;
        end
    end
end
end

```

```

        end
    end

    if (IN(i,j-1) ~= IN(i-1,j+1));
        if (IN(i,j-1) ~= 0);
            IN(i,j)=IN(i,j-1);

            if (IN(i-1,j+1) ~= 0);
                if IN(i-1,j+1)<IN(i,j-1);
                    R(s,t) = IN(i-1,j+1);
                    R(s,t+1) = IN(i,j-1);
                    s=s+1;
                end
                if IN(i-1,j+1)>IN(i,j-1);
                    R(s,t) = IN(i,j-1);
                    R(s,t+1) = IN(i-1,j+1);
                    s=s+1;
                end
            end
        end
    end
    if IN(i-1,j+1)~=0;
        IN(i,j)=IN(i-1,j+1);

        if (IN(i-1,j+1) ~= 0);
            if IN(i-1,j+1)<IN(i,j-1);
                R(s,t) = IN(i-1,j+1);
                R(s,t+1) = IN(i,j-1);
                s=s+1;
            end
            if IN(i-1,j+1)>IN(i,j-1);
                R(s,t) = IN(i,j-1);
                R(s,t+1) = IN(i-1,j+1);
                s=s+1;
            end
        end
    end
end

if (IN(i-1,j-1) ~= IN(i-1,j));
    if (IN(i-1,j-1) ~=0);
        IN(i,j) =IN(i-1,j-1);
        if (IN(i-1,j)~=0);
            if IN(i-1,j)<IN(i-1,j-1);
                R(s,t) = IN(i-1,j);
                R(s,t+1) = IN(i-1,j-1);
                s=s+1;
            end
            if IN(i-1,j)>IN(i-1,j-1);
                R(s,t) = IN(i-1,j-1);
                R(s,t+1) = IN(i-1,j);
                s=s+1;
            end
        end
    end
end
if IN(i-1,j)~=0;
    IN(i,j) = IN(i-1,j);
end

```

```

        if (IN(i-1,j)~=0);
            if IN(i-1,j)<IN(i-1,j-1);
                R(s,t) = IN(i-1,j);
                R(s,t+1) = IN(i-1,j-1);
                s=s+1;
            end
            if IN(i-1,j)>IN(i-1,j-1);
                R(s,t) = IN(i-1,j-1);
                R(s,t+1) = IN(i-1,j);
                s=s+1;
            end
        end
    end
end

if(IN(i-1,j-1) ~= IN(i-1,j+1));
    if IN(i-1,j-1) ~= 0;
        IN(i,j) = IN(i-1,j-1);
        if(IN(i-1,j+1)~=0);
            if IN(i-1,j+1)<IN(i-1,j-1);
                R(s,t) = IN(i-1,j+1);
                R(s,t+1) = IN(i-1,j-1);
                s=s+1;
            end
            if IN(i-1,j+1)>IN(i-1,j-1);
                R(s,t) = IN(i-1,j-1);
                R(s,t+1) = IN(i-1,j+1);
                s=s+1;
            end
        end
    end
end
if IN(i-1,j+1)~=0;
    IN(i,j) = IN(i-1,j+1);
    if(IN(i-1,j+1)~=0);
        if IN(i-1,j+1)<IN(i-1,j-1);
            R(s,t) = IN(i-1,j+1);
            R(s,t+1) = IN(i-1,j-1);
            s=s+1;
        end
        if IN(i-1,j+1)>IN(i-1,j-1);
            R(s,t) = IN(i-1,j-1);
            R(s,t+1) = IN(i-1,j+1);
            s=s+1;
        end
    end
end
end
end

if(IN(i-1,j)~=IN(i-1,j+1));
    if(IN(i-1,j)~=0);
        IN(i,j) = IN(i-1,j);

        if(IN(i-1,j+1)~=0);
            if IN(i-1,j+1)<IN(i-1,j);
                R(s,t) = IN(i-1,j+1);
                R(s,t+1) = IN(i-1,j);
            end
        end
    end
end

```

```

        s=s+1;
    end
    if IN(i-1,j+1)>IN(i-1,j);
        R(s,t) = IN(i-1,j);
        R(s,t+1) = IN(i-1,j+1);
        s=s+1;
    end
end
end
end
if IN(i-1,j+1)~=0;
    IN(i,j) = IN(i-1,j+1);
    if(IN(i-1,j+1)~=0);
        if IN(i-1,j+1)<IN(i-1,j);
            R(s,t) = IN(i-1,j+1);
            R(s,t+1) = IN(i-1,j);
            s=s+1;
        end
        if IN(i-1,j+1)>IN(i-1,j);
            R(s,t) = IN(i-1,j);
            R(s,t+1) = IN(i-1,j+1);
            s=s+1;
        end
    end
end
end
end
end
end

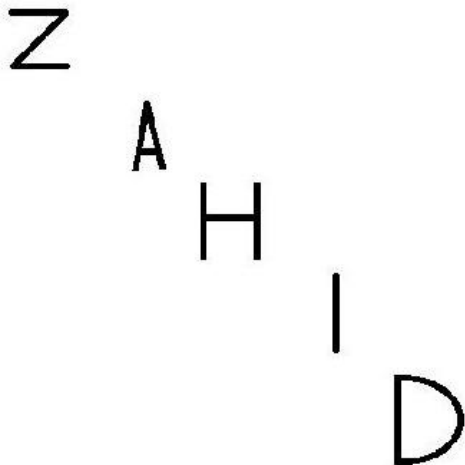
%Making output image from equivalence table
IO = IN;
z=size(R); %finding size of Equivalence table
y=z(1,1); % y represent number of rows in equivalence table
for o = y:-1:1; %starting loop from end of equivalence table
    if R(o,1)~=0 && R(o,2)~=0; %checking the values in equivalence table,
when both colomns are non zero
        for i=1:soi(1,1); %number of rows in image
            for j=1:soi(1,2); %number of colomns in image
                if IO(i,j)== R(o,2); %when value in output image pixel is
equal to 2nd colomn of equivalence table,
                    IO(i,j) = R(o,1); % then replace it with 1st colomn
                end
            end
        end
    end
end
end

%assigning different colour to different objects
a=1;
for i=1:soi(1,1)
    for j=1:soi(1,2)
        if IO(i,j)>a;
            a=IO(i,j);
        end
    end
end
end

```

```
z=255/a;  
I3=I0;  
for c=1:a  
    for i=1:soi(1,1);  
        for j=1:soi(1,2);  
            if I0(i,j)== c;  
                I3(i,j,1) = 255-c*z;  
                I3(i,j,2) = c*z/2;  
                I3(i,j,3) = 2*c*z;  
            end  
        end  
    end  
end  
  
%convering to 8 bits unsigned integer  
I3 = uint8(I3);  
subplot(1,2,1);  
imshow(I);  
title('Input Image');  
subplot(1,2,2);  
imshow(I3);  
title('Output Image');
```

Input Image



Z  
A  
H  
I  
D

Output Image

