Table of Contents

Question 3

```
tic;
```

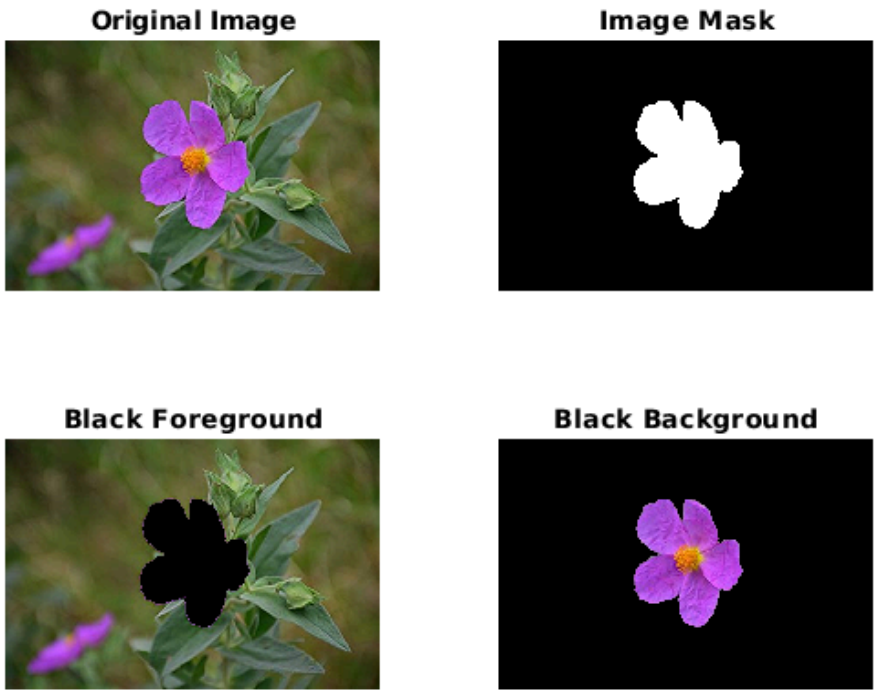
(a) Obtaining Image Mask

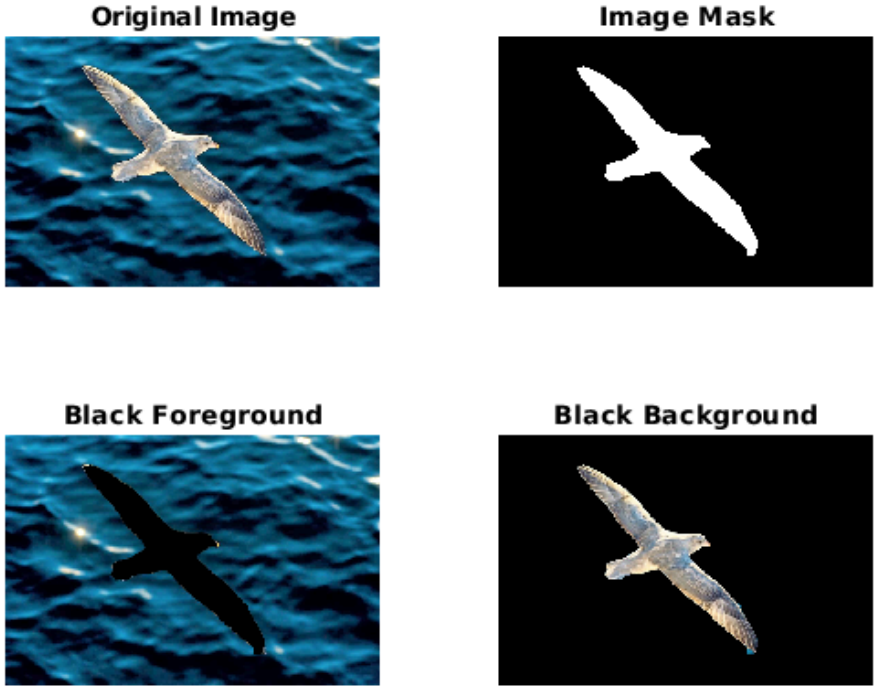
Algorithm: 1) Do Mean Shift Segmentation on the image \\ 2) From the segmented image identify which color component(out of RGB) is maximum or minimum in foreground region and accordingly apply the color component filter (conditions on R,G,B components of each pixel) such that foreground segment gets bright and background dark \\ 3) Alongwith foreground if some pixels of background are also bright then manually identify them and set them to zero \\ In Flower image foreground region is purple(Blue+red) color (more of blue component) and the little region between petals is yellow colour(more towards red component)\\ So thats how we can apply the color component filter to get the mask\\ In Bird image, background region has very less values for the red component, hence accordingly filter is applied

```
%  Flower image
flower_img=imread("flower.jpg");
[m,n,~]=size(flower_img);

flower_mask=load("MSSFlower.mat").output_img;
% MSSFlower is the Mean shift segmented image of Flower
flower_mask=255*flower_mask;

for i=1:m
    for j=1:n
        im_comp=flower_mask(i,j,:);
        im_comp=reshape(im_comp,1,3);
        if im_comp(1,3)>150||im_comp(1,1)>150
            % im_comp(1,3) is blue colour intensity of pixel (i,j)
            % and im_comp(1,1) is for red colour
            flower_mask(i,j,:)=[255,255,255];
        else
            flower_mask(i,j,:)=0;
        end
    end
end

% Manually identified the indices of the background region pixels
% which are bright, they are [200:283,1:120]
for i=200:m
```

```
    for j=1:120
        flower_mask(i,j,:)=0;
    end
end
flower_mask=flower_mask(:,:,1)/255;

% For Black background and foreground image
bl_backg=imread("flower.jpg");
bl_foreg=imread("flower.jpg");
for i=1:m
    for j=1:n
        if flower_mask(i,j)==0
            bl_backg(i,j,:)=0;
        else
            bl_foreg(i,j,:)=0;
        end
    end
end


figure,subplot(2,2,1),imshow(flower_img)
title('Original Image')
subplot(2,2,2),imshow(flower_mask)
title('Image Mask')
subplot(2,2,3),imshow(bl_foreg)
title('Black Foreground')
subplot(2,2,4),imshow(bl_backg)
title('Black Background')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Bird Image
bird_img=imread("bird.jpg");
bird_img=imresize(bird_img,0.5);

bird_mask=load("MSSBird.mat").output_img;
[m,n,c]=size(bird_mask);
bird_mask=255*bird_mask;
for i=1:m
    for j=1:n
        im_comp=bird_mask(i,j,:);
        im_comp=reshape(im_comp,1,3);
        if im_comp(1,1)>80 %im_comp(1,1) is the red colour component
 of (i,j)
            bird_mask(i,j,:)=[255,255,255];
        else
            bird_mask(i,j,:)=0;
        end
    end
end

% Indices of the background region which are bright are
% in between [130:155,90:125]
for i=130:155
    for j=90:125
```

```
        bird_mask(i,j,:)=0;
    end
end
bird_mask=bird_mask(:,:,1)/255;

% for black foreground and background
bl_backgb=bird_img;
bl_foregb=bird_img;
for i=1:m
    for j=1:n
        if bird_mask(i,j)==0
            bl_backgb(i,j,:)=0;
        else
            bl_foregb(i,j,:)=0;
        end
    end
end

figure,subplot(2,2,1),imshow(bird_img)
title('Original Image')
subplot(2,2,2),imshow(bird_mask)
title('Image Mask')
subplot(2,2,3),imshow(bl_foregb)
title('Black Foreground')
subplot(2,2,4),imshow(bl_backgb)
title('Black Background')
```





(b) Background Blurring

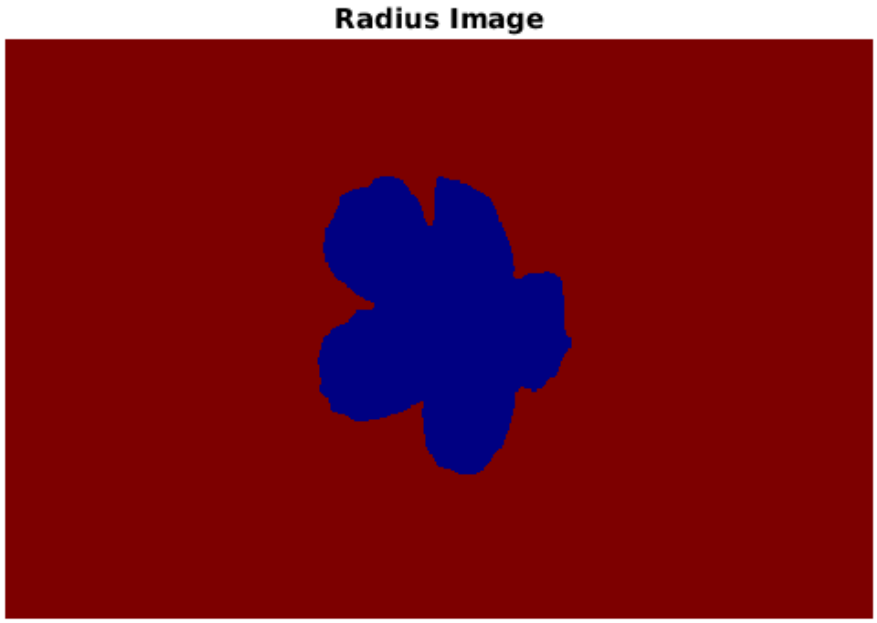
inputs of mySpatiallyVaryingKernel function are input image, mask image, spatial bandwidth parameter(hs) and alpha value outputs are the blurred image, image of radii, and blurring kernels for different values of radius

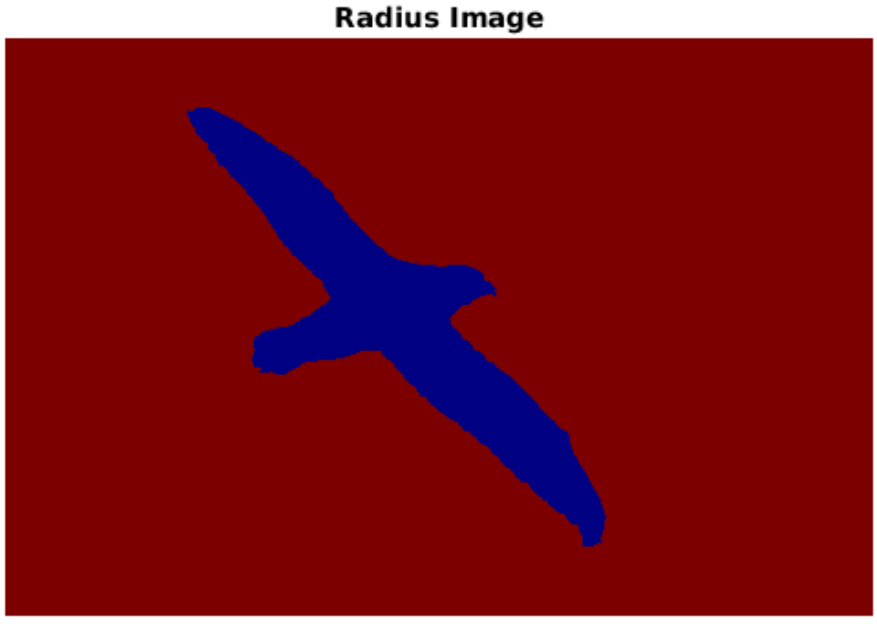
```
% Flower Image
[finalIm_fl,radIm_fl,blrKern_fl]=mySpatiallyVaryingKernel(flower_img,flower_mask,8
% Bird Image
[finalIm_br,radIm_br,blrKern_br]=mySpatiallyVaryingKernel(bird_img,bird_mask,10,40
```

(c) Image of Radius values

Flower Image

```
figure,imshow(radIm_fl)
title('Radius Image')
colormap("jet")

% Bird Image
figure,imshow(radIm_br)
title('Radius Image')
colormap("jet")
```


Radius Image

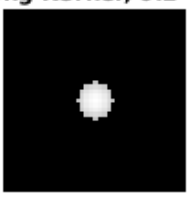

Radius Image

(d) Image of blurring kernels at d_p=(0.2,0.4,0.6,0.8,1)*alpha

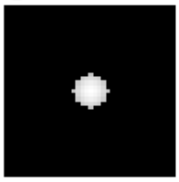
Flower Image

```
figure
subplot(3,2,1),imshow(blrKern_fl(:,:,1))
title('Blurring Kernel, 0.2*alpha')
subplot(3,2,2),imshow(blrKern_fl(:,:,1))
title('Blurring Kernel, 0.4*alpha')
subplot(3,2,3),imshow(blrKern_fl(:,:,1))
title('Blurring Kernel, 0.6*alpha')
subplot(3,2,4),imshow(blrKern_fl(:,:,1))
title('Blurring Kernel, 0.8*alpha')
subplot(3,2,5),imshow(blrKern_fl(:,:,1))
title('Blurring Kernel, alpha')

%%%%%%%%%%%%%%%%
% Bird Image
figure
subplot(3,2,1),imshow(blrKern_br(:,:,1))
title('Blurring Kernel, 0.2*alpha')
subplot(3,2,2),imshow(blrKern_br(:,:,1))
title('Blurring Kernel, 0.4*alpha')
subplot(3,2,3),imshow(blrKern_br(:,:,1))
title('Blurring Kernel, 0.6*alpha')
subplot(3,2,4),imshow(blrKern_br(:,:,1))
title('Blurring Kernel, 0.8*alpha')
subplot(3,2,5),imshow(blrKern_br(:,:,1))
title('Blurring Kernel, alpha')
```

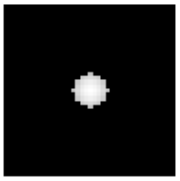
Blurring Kernel, 0.2*alpha



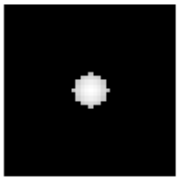
Blurring Kernel, 0.4*alpha



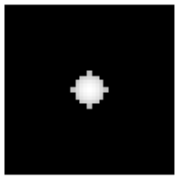
Blurring Kernel, 0.6*alpha



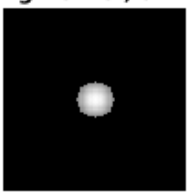
Blurring Kernel, 0.8*alpha



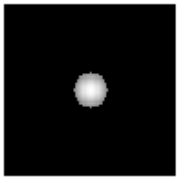
Blurring Kernel, alpha



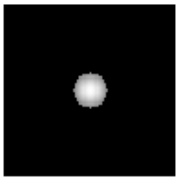
Blurring Kernel, 0.2*alpha



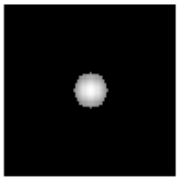
Blurring Kernel, 0.4*alpha



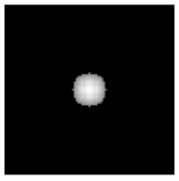
Blurring Kernel, 0.6*alpha



Blurring Kernel, 0.8*alpha



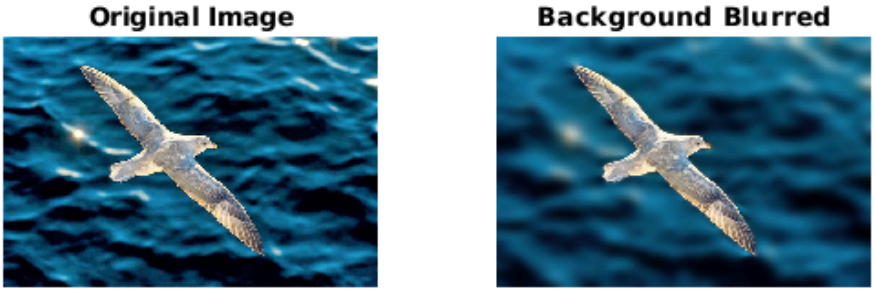
Blurring Kernel, alpha



(e) Display of background blurred images

Flower Image

```
figure
subplot(1,2,1),imshow(flower_img)
title('Original Image')
subplot(1,2,2),imshow(finalIm_fl)
title('Background Blurred')

% Bird Image
figure
subplot(1,2,1),imshow(bird_img)
title('Original Image')
subplot(1,2,2),imshow(finalIm_br)
title('Background Blurred')

toc;

Elapsed time is 62.990501 seconds.
```



Published with MATLAB® R2020b