Question 1. Implement an android function to capture a focal stack (5 Points)

Answer:

Android Studio Codes

```
public void captureFocalStack(View v) {
        //TODO:hw5
        //getting min and max focusdistance
       //Range<Long> focalRange = characteristics.get(CameraCharacteristics.);
        float minimumLens =
characteristics.get(CameraCharacteristics.LENS_INFO_MINIMUM_FOCUS_DISTANCE);
        float maximumLens =
characteristics.get(CameraCharacteristics.LENS_INFO_HYPERFOCAL_DISTANCE);
        Log.e(TAG, "minimumLens: " + minimumLens);
        Log.e(TAG, "maxmimumLens: " + maximumLens);
        //TODO:hw5
        //setting previous lens to be min or max focus distance. (guess which one it is!)
        float prev_focus = maximumLens;
        Log.e(TAG, "in captureFocalStack");
        //check if capture session is null
        if (mCaptureSession != null) {
            Log.e(TAG, "prevLens: " + prev_focus);
            //TODO: check if focus distance after changing is in range
            while (prev_focus * 1.5 < minimumLens) {
                //sleep system clock for 20 ms
                SystemClock.sleep(20);
                Log.e(TAG, "in captureFocalStack while loop");
                try {
                    //TODO: set current focus to be 1.5 * previous focus
                    float curr_focus = (float)1.5 * prev_focus;
                    focuses.add(curr_focus);
                    //build requester
                    CaptureRequest.Builder requester =

mCameraDevice.createCaptureRequest(mCameraDevice.TEMPLATE_STILL_CAPTURE);
                    //TODO: turn off auto focus mode for requester
                    requester.set(CaptureRequest.CONTROL_AF_MODE,
CaptureRequest.CONTROL_AF_MODE_OFF);
                    //add surface as target in requester
                    requester.addTarget(mCaptureBuffer.getSurface());
                    //TODO: set current focus to requester
                    requester.set(CaptureRequest.LENS_FOCUS_DISTANCE, curr_focus);
                    //set previous focus = current focus
                    prev_focus = curr_focus;
```

```
                    try {
                        // This handler can be null because we aren't actually attaching any callback
                        //make capture session

                        mCaptureSession.capture(requester.build(), /*listener*/null, /*handler*/null);
                    } catch (CameraAccessException ex) {
                        Log.e(TAG, "Failed to file actual capture request", ex);
                    }
                } catch (CameraAccessException ex) {
                    Log.e(TAG, "Failed to build actual capture request", ex);
                }
            }
        } else {
            Log.e(TAG, "User attempted to perform a capture outside the session");
        }

}
```

I found the minimumLens is 0.67 and the maximumLens is 10. So just 6 pictures were saved and focus situations are bad. To solve the problem, I set different positions of the objects to get appropriate focus on them. Finally, I chose focus = 3.375, 5.0625 and 7.59375. Three images are as follows.

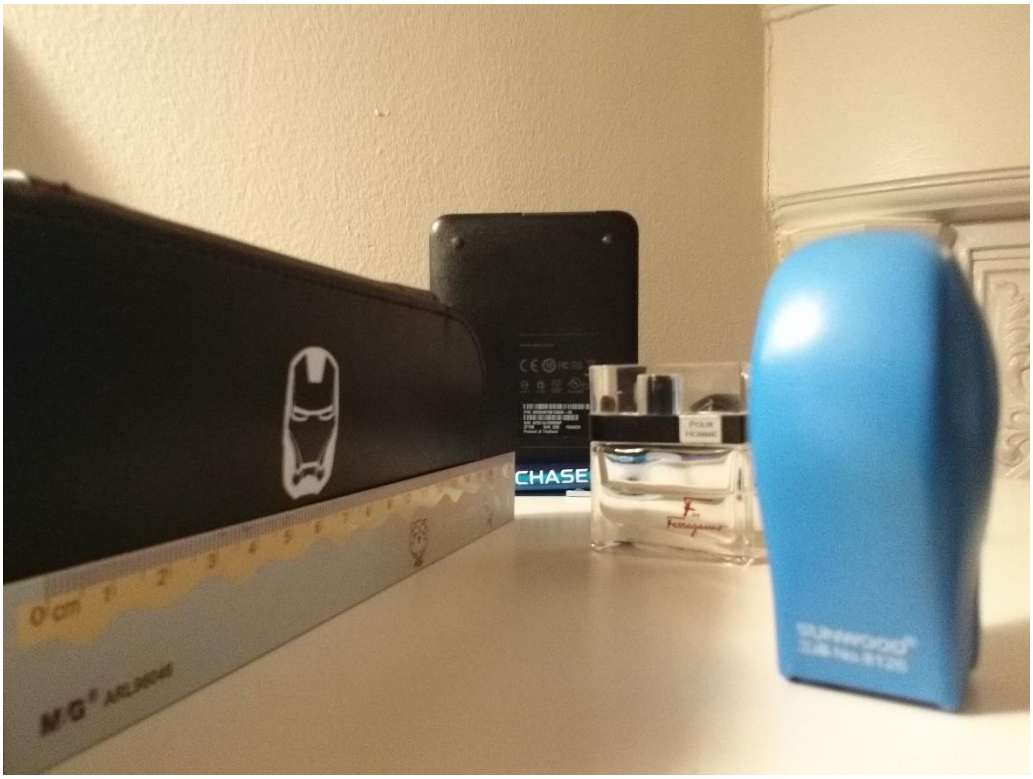

Figure 1. focus = 3.375

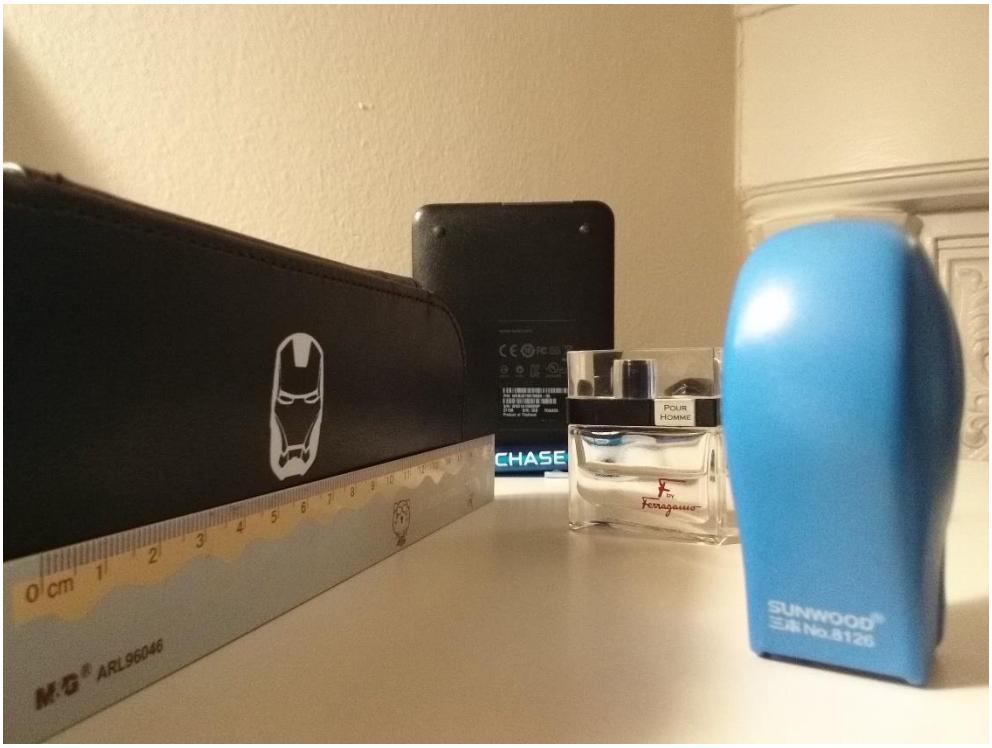

Figure 2. focus = 5.0625

Figure 3. focus= 7.59375

Question 2. Calibrate your focal stack (2 Points)

Answer:
When calibrating the three images, there is something wrong with the magnitude which makes the calibrating is

not accurate. However, I tried several parameters and found multiplying their focus by 1.5 can get almost perfect results of calibration. The results are as follows.

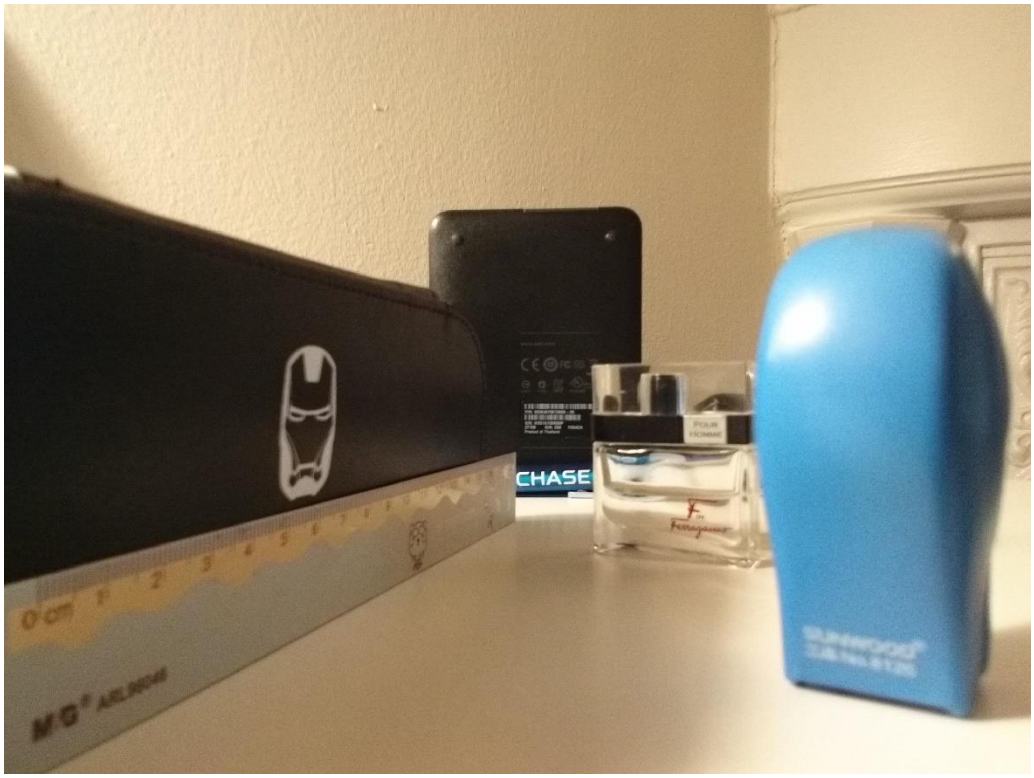

Figure 4. focus = 3.375, calibrated

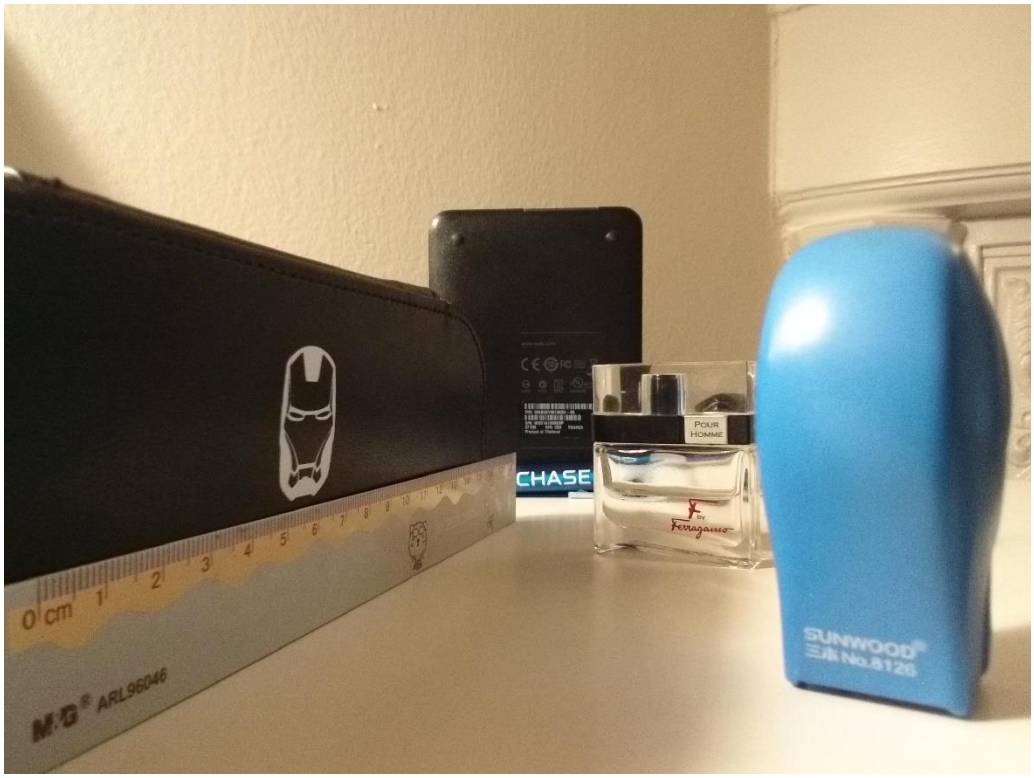

Figure 5. focus = 5.0625, calibrated

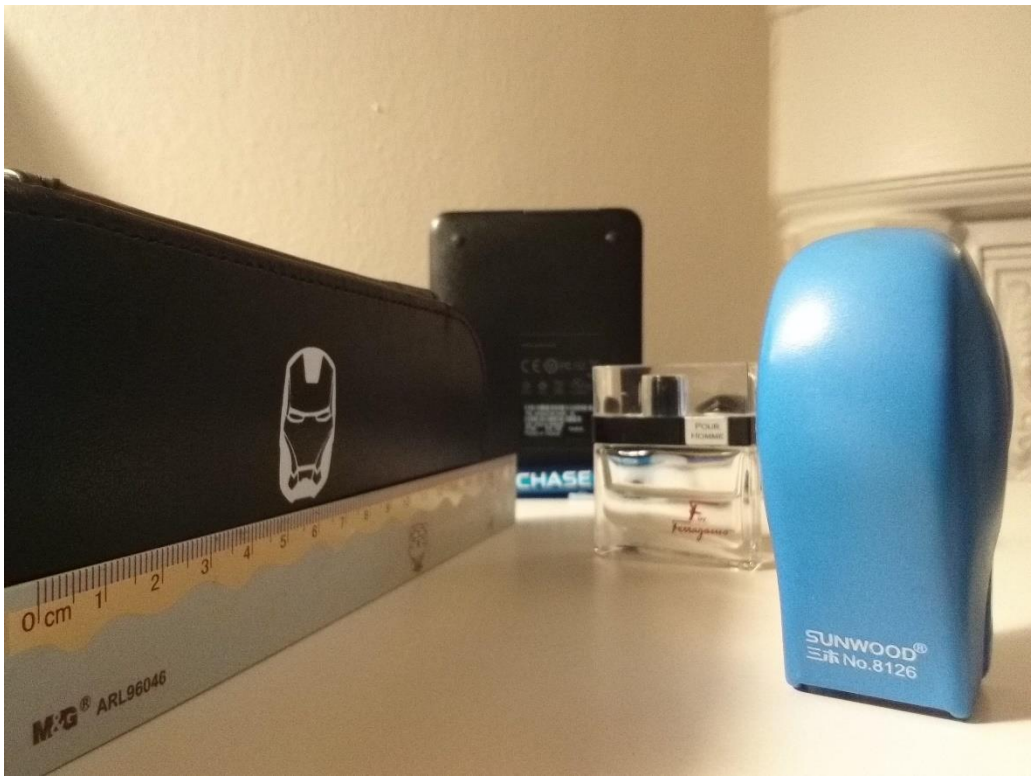

Figure 6. focus = 7.5975 calibrated

Matlab Codes

```
clear;
clc;
k=3;
Iori=cell(k,1);
U=zeros(k,1);
f=2.95;
Vkstack=[1.5^4;1.5^5;1.5^6];
for i=1:k
    U(i)=1/(339-Vkstack(i));
end
for i=1:k
    imgname=strcat('D:\Courses Files_2015_Fall\Introduction to Computational
Photography\HW5\',num2str(i),'.jpg');
    Iori{i}=im2double(imread(imgname));
end
Icalib=calibrate(Iori,k,U);
function Iout=calibrate(Iin,k,U)
Iout=cell(k,1);
for i=1:k
    red=Iin{i}(:,:,1);
    green=Iin{i}(:,:,2);
    blue=Iin{i}(:,:,3);
```

```
    [m,n]=size(red);
    mi=U(i)/U(k);
    for a=1:m
        for b=1:n
            if a<=m/2
                a2=round(m/2-abs(a-m/2)*mi);
            end
            if a>m/2
                a2=round(m/2+abs(a-m/2)*mi);
            end
            if b<=n/2
                b2=round(n/2-abs(b-n/2)*mi);
            end
            if b>n/2
                b2=round(n/2+abs(b-n/2)*mi);
            end
            rcalib(a,b)=red(a2,b2);
            gcalib(a,b)=green(a2,b2);
            bcalib(a,b)=blue(a2,b2);
        end
    end
    Iout{i}(:,:,1)=rcalib;
    Iout{i}(:,:,2)=gcalib;
    Iout{i}(:,:,3)=bcalib;
    imgname=strcat('calibrated',num2str(i),'.jpg');
    imwrite(Iout{i},imgname,'jpeg');
end
```

Question 3. Compute a depth map from the focal stack (3 Points)

Answer:

K was set as 1, 2,3,5 and 7. Results are as follows.

Figure 7. focus = 3.375 depth map, K=1

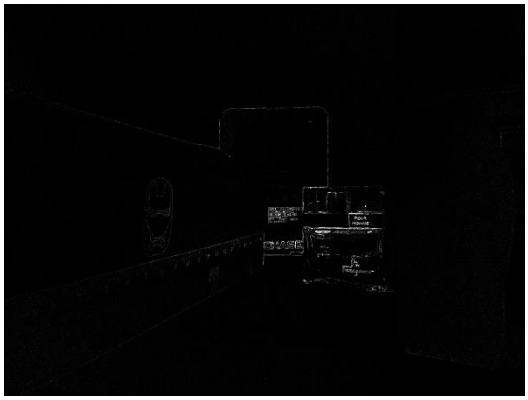

Figure 8. focus = 5.0625 depth map, K=1

Figure 9. focus = 7.5975 depth map, K=1

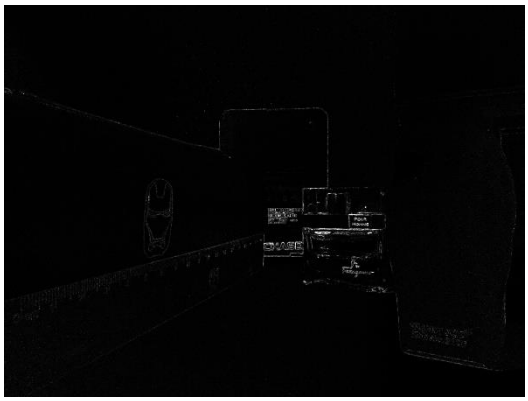

Figure 10. depth map combined, K=1

Figure 11. focus = 3.375 depth map, K=2

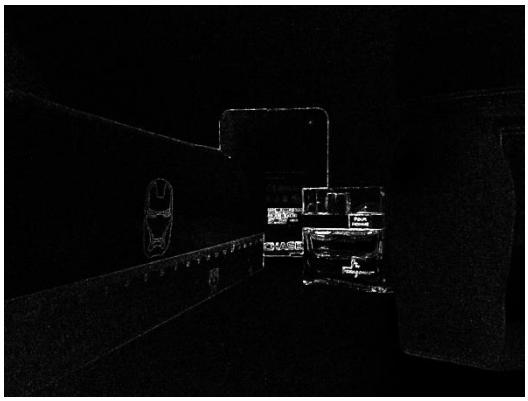

Figure 12. focus = 5.0625 depth map, K=2

Figure 13. focus = 7.5975 depth map, K=2

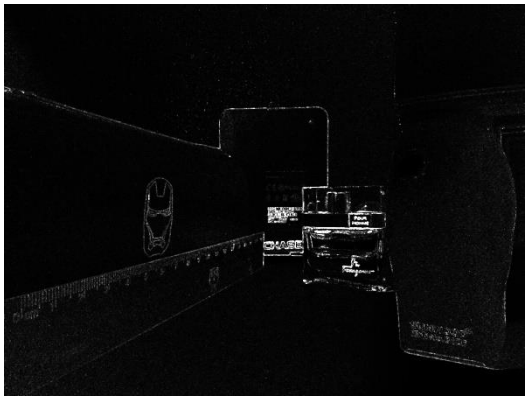

Figure 14. depth map combined, K=2

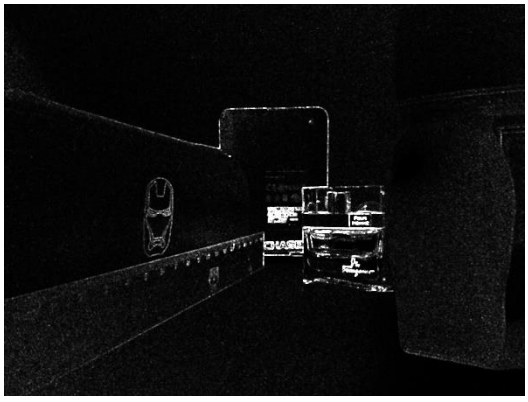

Figure 15. focus = 3.375 depth map, K=3

Figure 16. focus = 5.0625 depth map, K=3

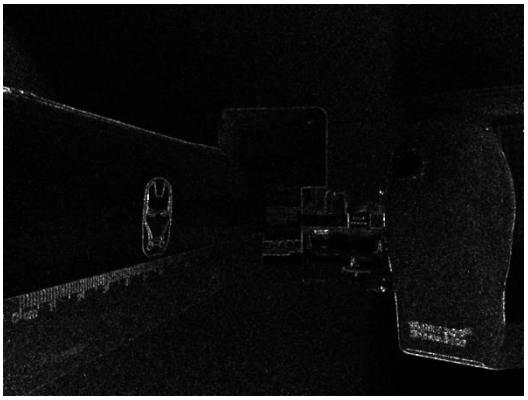

Figure 17. focus = 7.5975 depth map, K=3

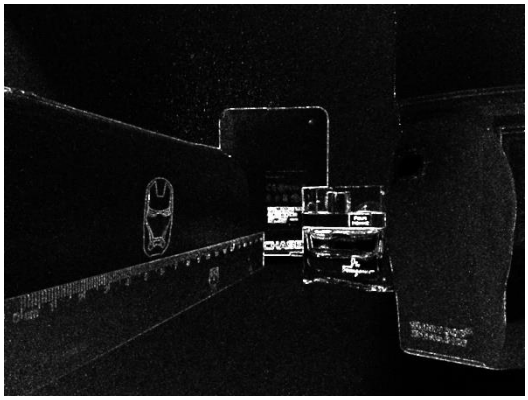

Figure 18. depth map combined, K=3

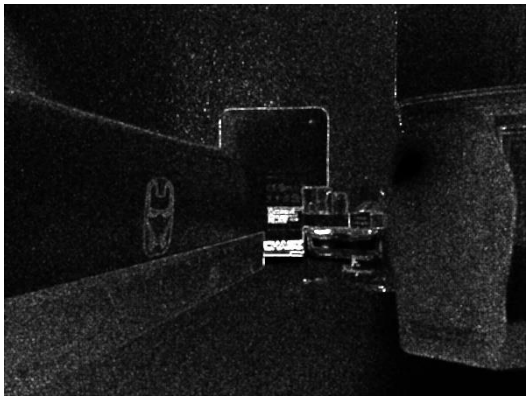

Figure 19. focus = 3.375 depth map, K=5

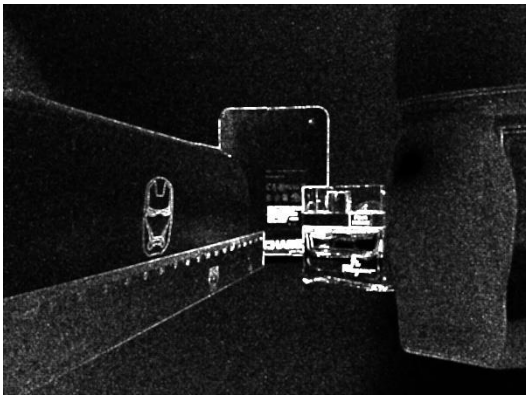

Figure 20. focus = 5.0625 depth map, K=5

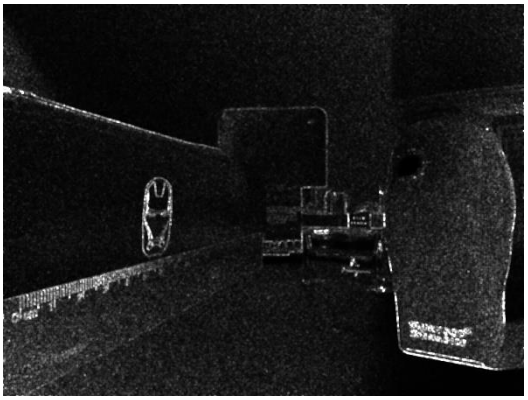

Figure 21. focus = 7.5975 depth map, K=5

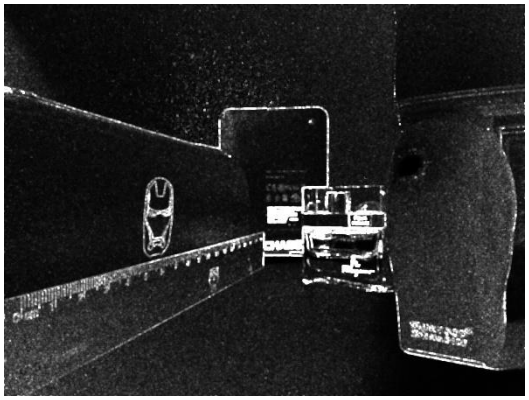

Figure 22. depth map combined, K=5

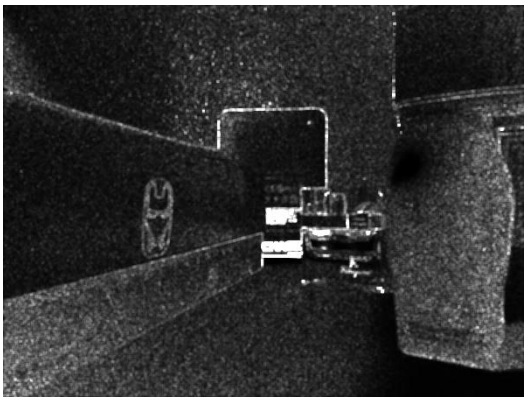

Figure 23. focus = 3.375 depth map, K=7

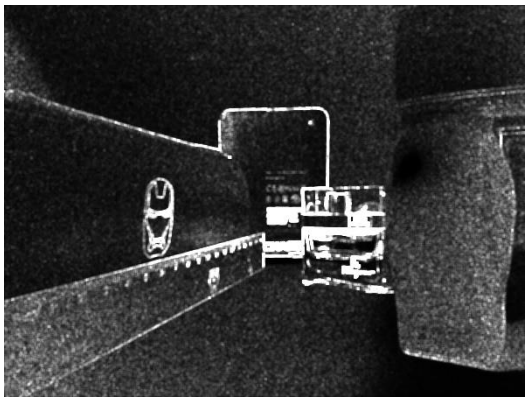

Figure 24. focus = 5.0625 depth map, K=7

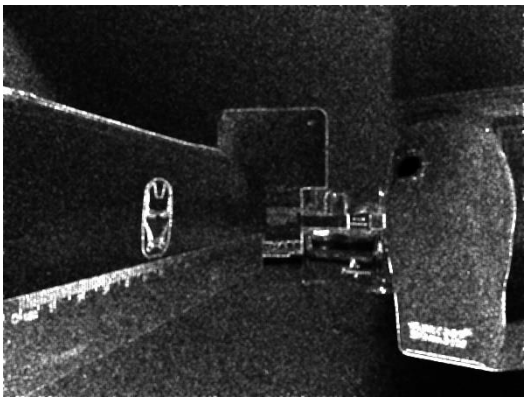

Figure 25. focus = 7.5975 depth map, K=7

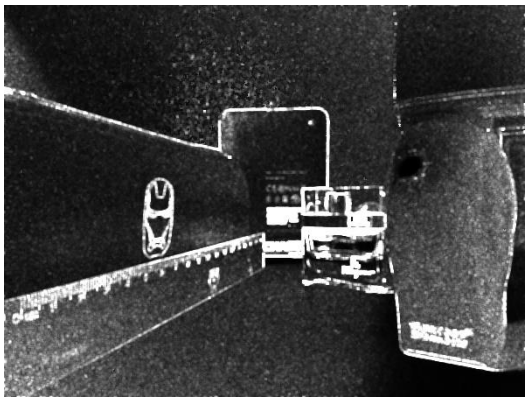

Figure 26. depth map combined, K=7

Matlab Codes

```
Igray=graytransformation(Icalib,k);
Ilaplacian=Laplacian(Igray,k);
IM=recover(Ilaplacian,k,5);
[Imap,Icomb]=compare(IM);
function Iout=graytransformation(Iin,k)
Iout=cell(k,1);
for i=1:k
    Iout{i}=rgb2gray(Iin{i});
end
function Iout=Laplacian(Iin,k)
Iout=cell(k,1);
mask=[1 4 1;
    4 -20 4
    1 4 1]/6;
for i=1:k
    Iout{i}=imfilter(Iin{i},mask,'replicate');
    Iout{i}=Iout{i}.*Iout{i};
end
function Iout=recover(Iin,k,K)
```

```
Iout=cell(k,1);
[m,n]=size(Iin{1});
for i=1:k
    Iout{i}=zeros(m,n);
end
for a=1:k
    for b=1:m
        for c=1:n
            for i=-K:K
                for j=-K:K
                    d=b+i;
                    e=c+j;
                    while d<=0
                        d=d+1;
                    end
                    while d>m
                        d=d-1;
                    end
                    while e<=0
                        e=e+1;
                    end
                    while e>n
                        e=e-1;
                    end
                    Iout{a}(b,c)=Iout{a}(b,c)+Iin{a}(d,e);
                end
            end
        end
    end
    imgname=strcat('M k= ',num2str(a),'.jpg');
    imwrite(Iout{a},imgname,'jpeg');
end
end
function [Iout1,Iout2]=compare(Iin)
[m,n]=size(Iin{1});
Iout1=zeros(m,n);
Iout2=Iin{1};
for i=1:m
    for j=1:n
        Iout1(i,j)=1;
        if Iin{2}(i,j)>Iin{1}(i,j)
            Iout1(i,j)=2;
            Iout2(i,j)=Iin{2}(i,j);
        end
        if Iin{3}(i,j)>Iin{2}(i,j)
```

```
                Iout1(i,j)=3;
                Iout2(i,j)=Iin{3}(i,j);
            end
        end
end
imwrite(Iout2,'depth index map.jpg','jpeg');
end
```

Question 4. Recover an all-focus image of the scene (5 Points)

Answer:

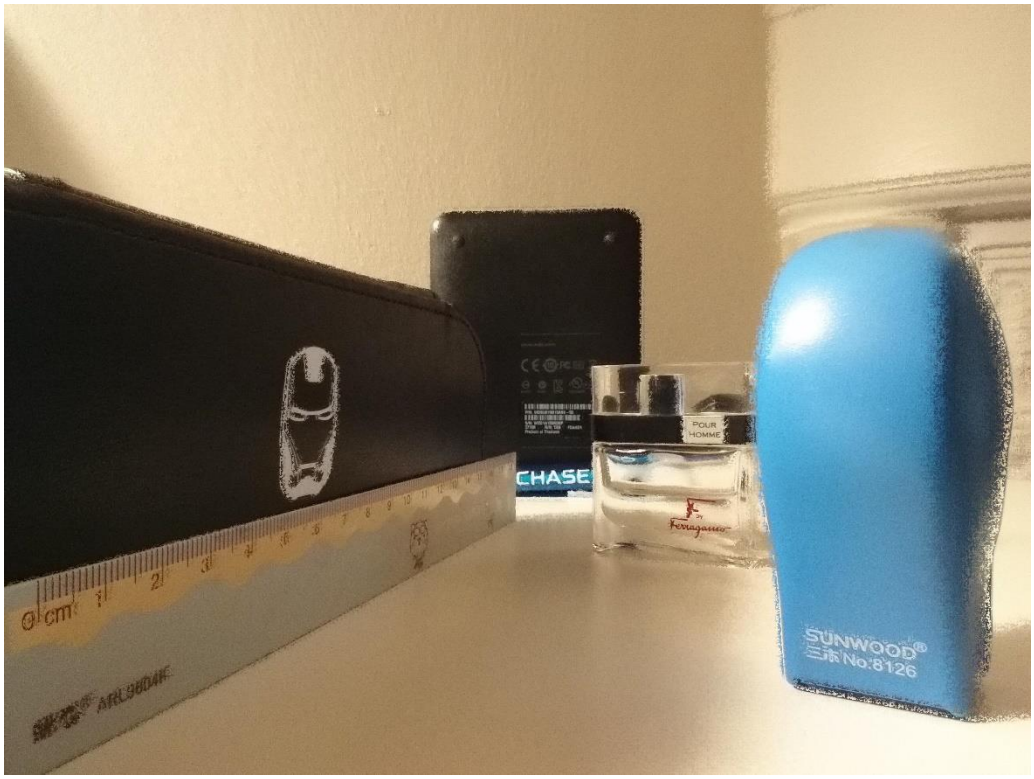

Figure 27. final result, K=1

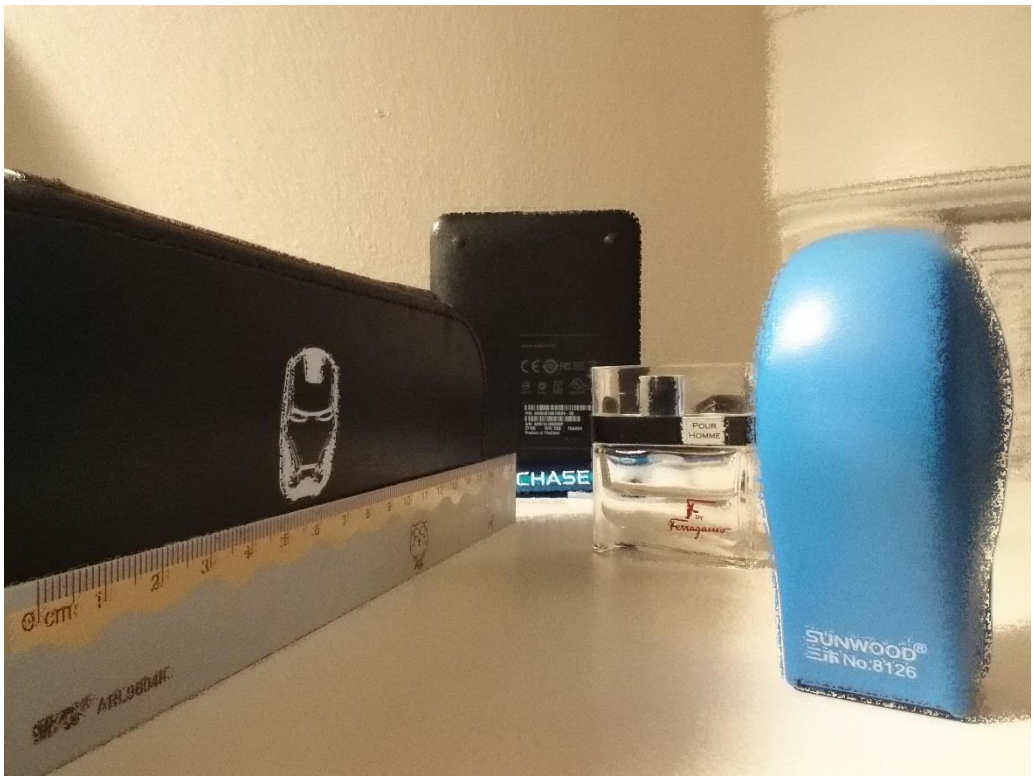

Figure 28. final result, K=2

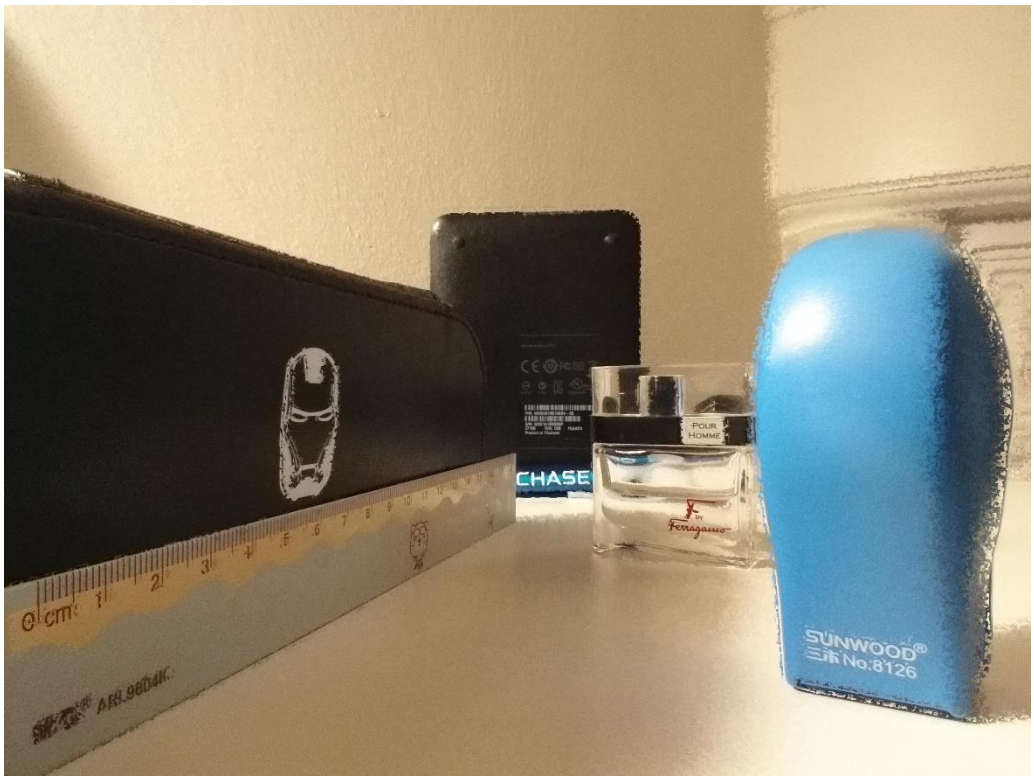

Figure 29. final result, K=3

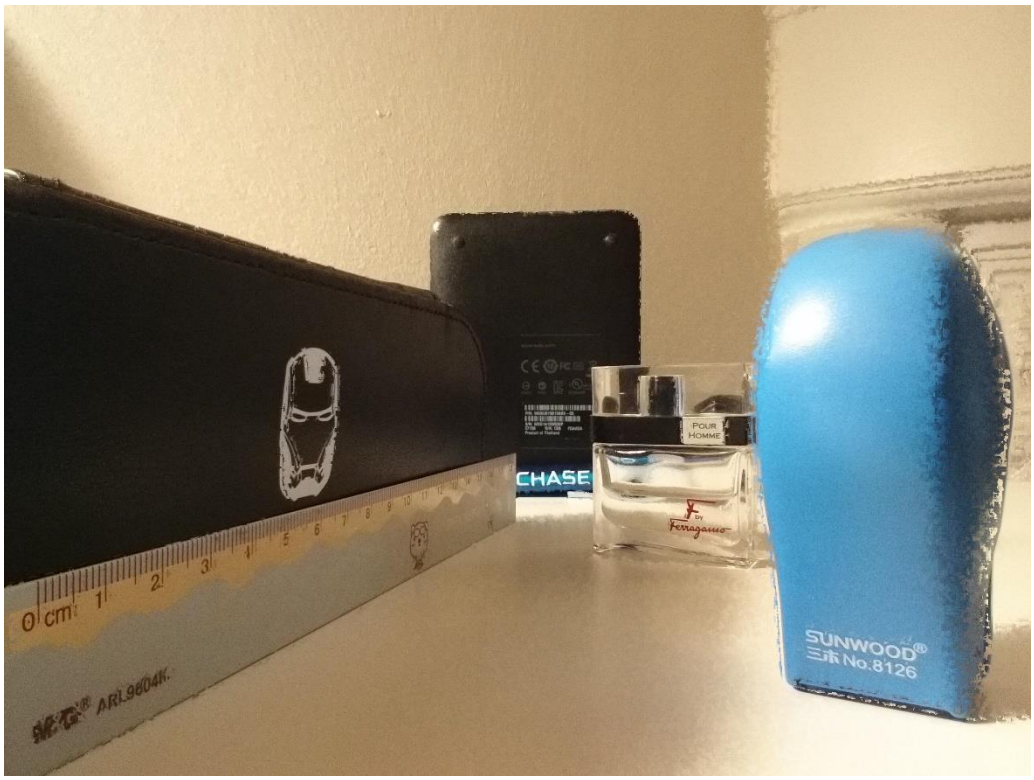

Figure 30. final result, K=5

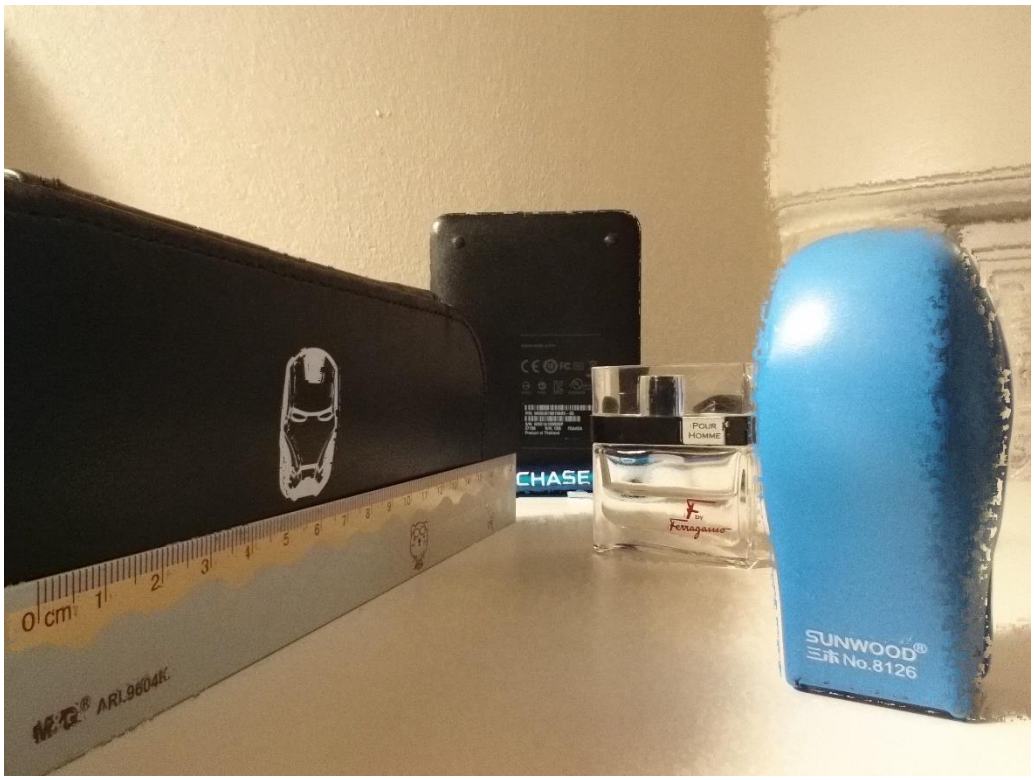

Figure 31. final result, K=7

Comparing the results we can see that as K increases, the all-in focus image get better and better.

Matlab Codes

```
[m,n]=size(Imap);
Ifinal=zeros(m,n,3);
for i=1:m
    for j=1:n
        Ifinal(i,j,1)=Iori{Imap(i,j)}(i,j,1);
        Ifinal(i,j,2)=Iori{Imap(i,j)}(i,j,2);
        Ifinal(i,j,3)=Iori{Imap(i,j)}(i,j,3);
    end
end
figure(),imshow(Ifinal);
imwrite(Ifinal,'Ifinal.jpg','jpeg');
```

Problems encountered:

1. From the Tablet, the minimumLens I got is larger than maximumLens. After discussing with professor Oliver and TA, I know focal distances are units of diopters.
2. Before calibration, we need multiply the focus distance by 1.5.