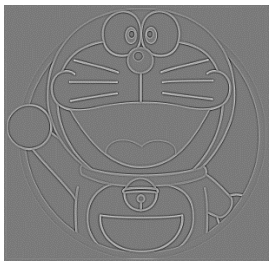
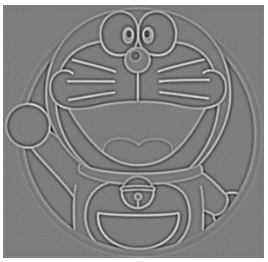
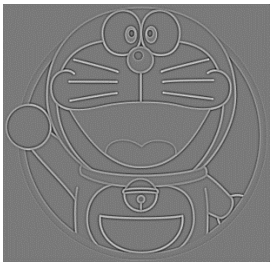
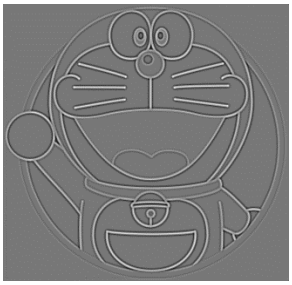
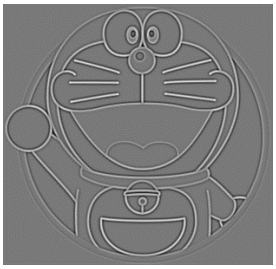
Computer Vision HW1 Report

Student ID: R09942171
Name: 黃繼綸

Part 1.

- **Visualize the detected corner for 1.png.**

	DoG Image (threshold = 5)		DoG Image (threshold = 5)
DoG1-1.png		DoG2-1.png	
DoG1-2.png		DoG2-2.png	
DoG1-3.png		DoG2-3.png	
DoG1-4.png		DoG2-4.png	

- **Use three thresholds (2, 5, 7) on 2.png and describe the difference.**

Threshold	Image with detected keypoints on 2.png
2	

5	
7	

From the comparison above, we can see that the key points will reduce when the threshold increases. If the threshold equals 2.0, the captured key points are very messy and catch a lot of edges. While the threshold equals 5.0, the captured key points have fewer edge points and are more representative. Further, when the threshold is 7.0, the points on the images are almost the most representative points

Part 2.

- **Report the cost for each filtered image.**

Gray Scale Setting	Cost (1.png)
cv2.COLOR_BGR2GRAY	1207799
R*0.0+G*0.0+B*1.0	1439568
R*0.0+G*1.0+B*0.0	1305961
R*0.1+G*0.0+B*0.9	1393620
R*0.1+G*0.4+B*0.5	1279697
R*0.8+G*0.2+B*0.0	1127913

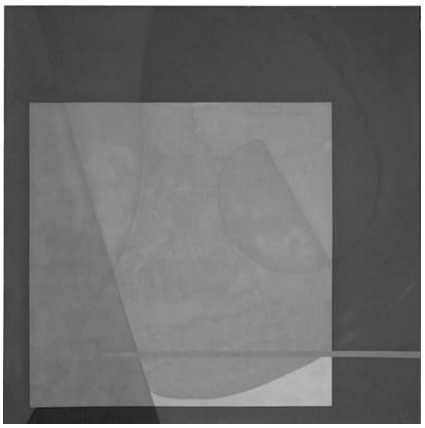
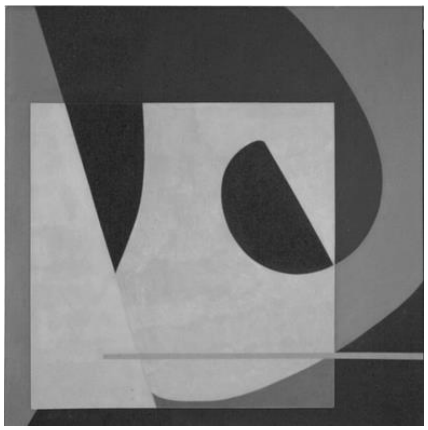
Gray Scale Setting	Cost (2.png)
cv2.COLOR_BGR2GRAY	183850
R*0.1+G*0.0+B*0.9	77883
R*0.2+G*0.0+B*0.8	86023
R*0.2+G*0.8+B*0.0	188019
R*0.4+G*0.0+B*0.6	128341
R*1.0+G*0.0+B*0.0	110862

- **Show original RGB image / two filtered RGB images and two grayscale images with highest and lowest cost.**

Original RGB image (1.png)	Filtered RGB image and Grayscale image of Highest cost	Filtered RGB image and Grayscale image of Lowest cost
		

From the grayscale image with the highest cost, We hardly differentiate the object and the something next to. While from the grayscale image with the lowest cost, we can simply differentiate the object and the something next to. That means the grayscale image generated by the weight of (0.8,0.2,0.0) is the best parameter to transform the color image to the grayscale image.

Original RGB image (2.png)	Filtered RGB image and Grayscale image of Highest cost	Filtered RGB image and Grayscale image of Lowest cost
	 	 

From the grayscale image with the highest cost, We hardly differentiate each object from it. While from the grayscale image with the lowest cost, we can even distinguish some objects from it. That means the grayscale image generated by the weight of (0.1,0.0,0.9) is the best parameter to transform the color image to the grayscale image.

- **Describe how to speed up the implementation of bilateral filter.**

I use the lookup table to speed up the computation of the range kernel. Because the range of $(T_p - T_q)^2$ is $0\sim255^2$, we can create the lookup table at first, as shown below:

```
self.LUT = np.exp(-np.arange(256) * np.arange(256) * scaleFactor_r)
```

In addition, the spatial kernel can also be created at first because the spatial kernel is constant if the sigma_s is determined, as shown below:

```
self.spatial_kernel = [[np.exp(-(i**2+j**2) * scaleFactor_s) \
                        for i in range(-(self.wndw_size // 2), (self.wndw_size +1) // 2)]\
                        for j in range(-(self.wndw_size // 2), (self.wndw_size +1) // 2)]
```