

## ASSIGNMENT – 02

### **How to Run a code:**

- Install :
  1. Matplotlib
  2. cv2
  3. Numpy
  4. Pandas
  5. scipy
- Open Jupyter Notebook and make sure that files and images, that are available in the same folder as jupyter notebook and then start running the code from the beginning.
- Input images present in the same folder as that of code files with question number.
- Run the cells sequentially.

**Key points :** Points in the image to match irrespective of rotation, translation, intensity and scale changes

### **1. Harris Corner Detection:**

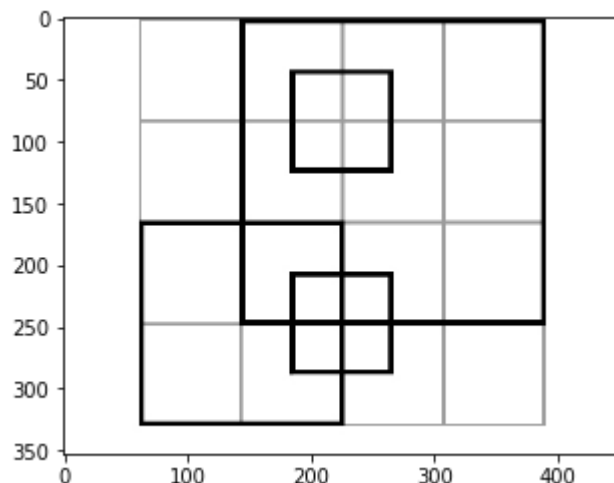
#### **Formula Used:**

$$R = \lambda_1 \lambda_2 - k \cdot (\lambda_1 + \lambda_2)^2 = \det(M) - k \cdot \text{tr}(M)^2$$

where  $k$  is an empirically determined constant;  $k \in [0.04, 0.06]$

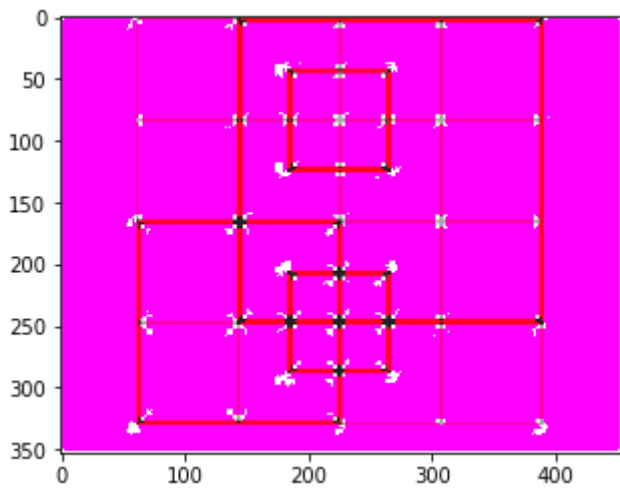
**First Image where value of  $k = 0.06$**

*Input Image:*

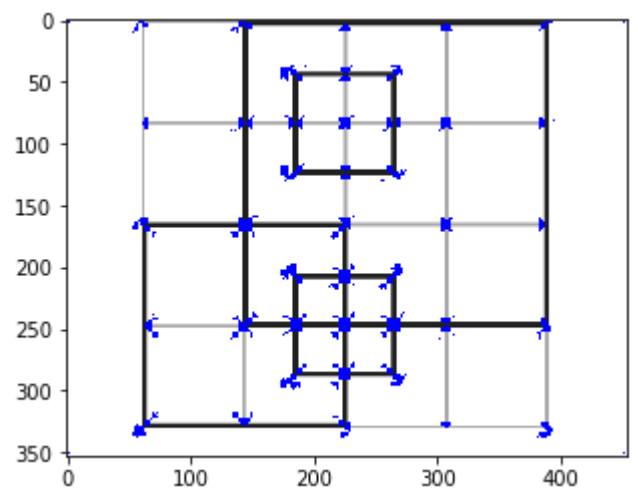


*Output Image:*

**“Edges”**

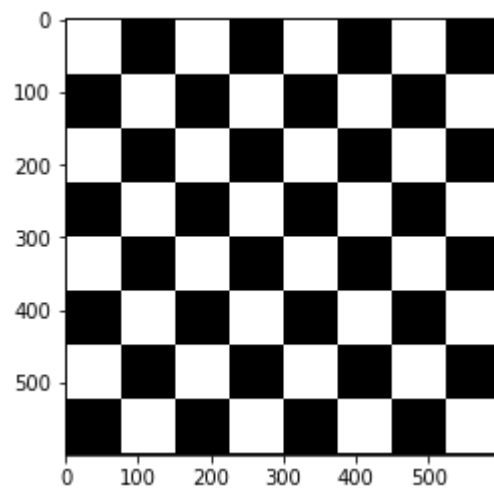


**“Corners”**



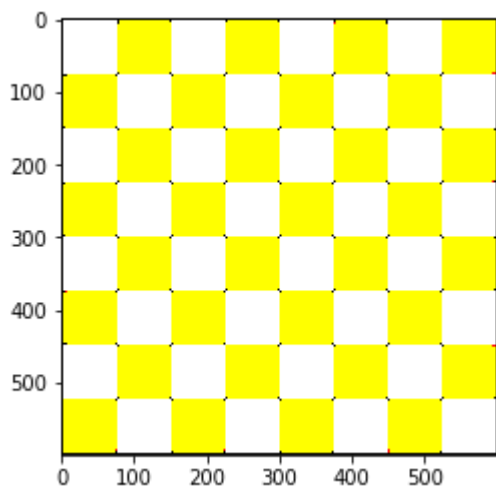
**Second image where value of  $k = 0.06$**

*Input Image*

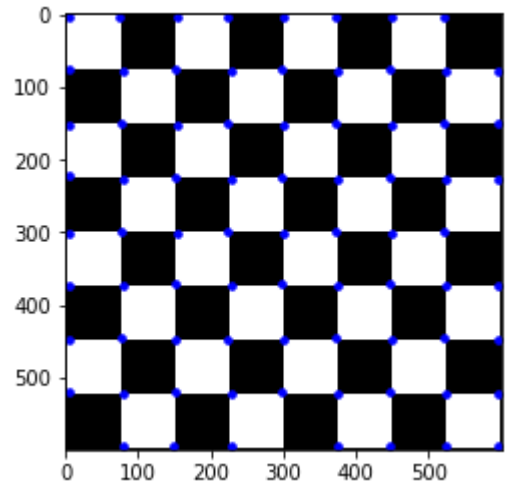


*Output Image*

**“Edges”**



**“Corners”**



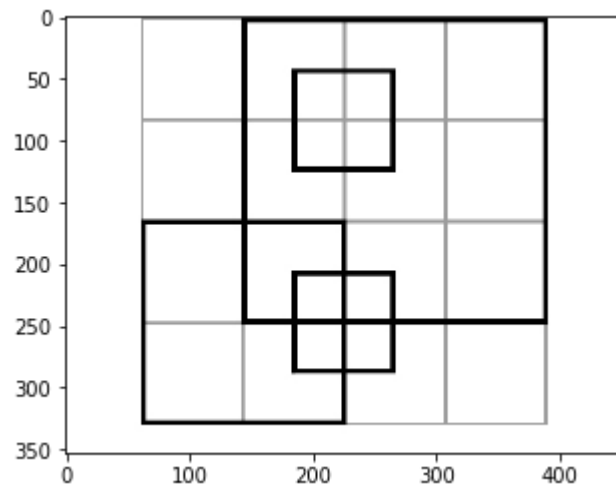
## 2. Shi Thomasi Corner Detection:

**Formula Used:**

$$R = \min(\lambda_1, \lambda_2)$$

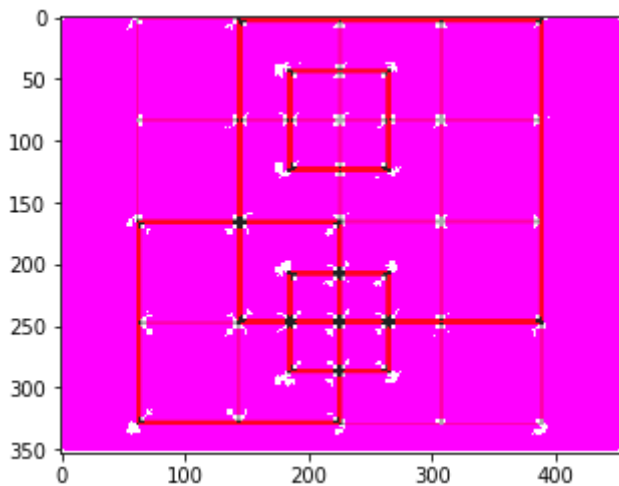
**First Image where value of k = 0.06**

Input Image

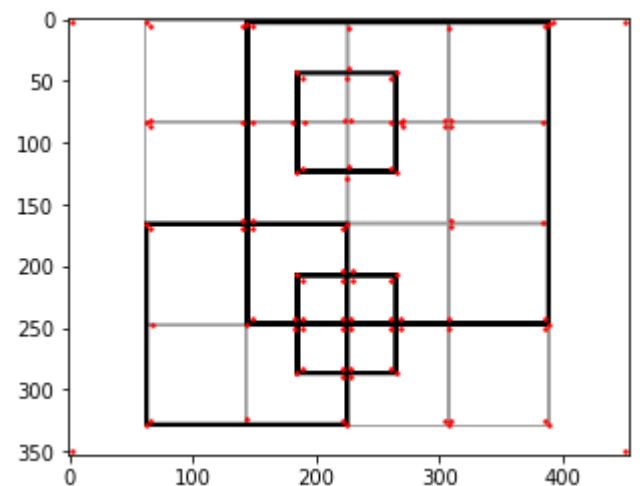


Output Image:

**“Edges”**

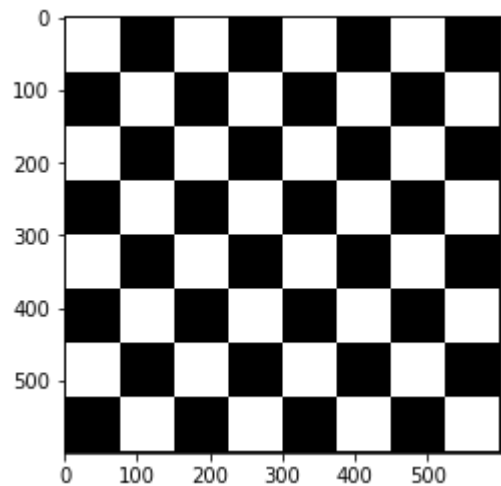


**“Corners”**



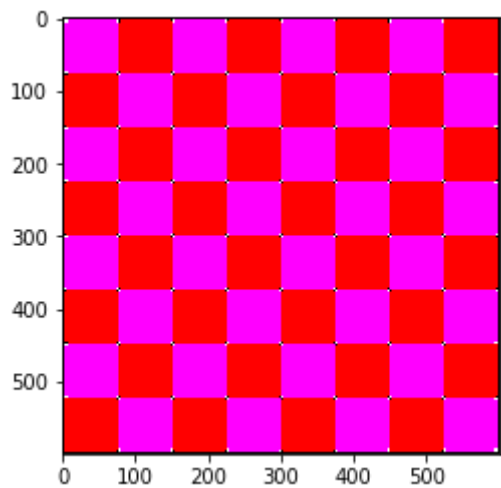
Second image where value of  $k = 0.06$

Input Image

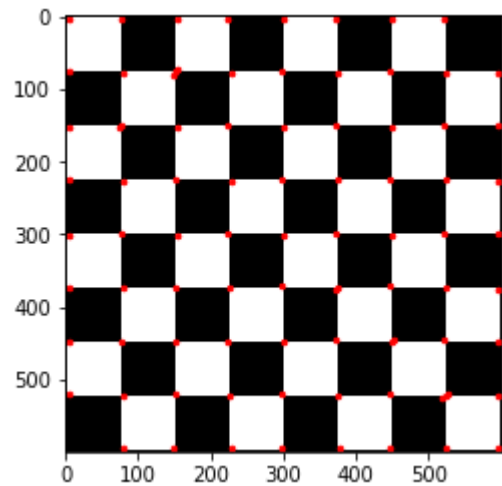


Output Image

“Edges”

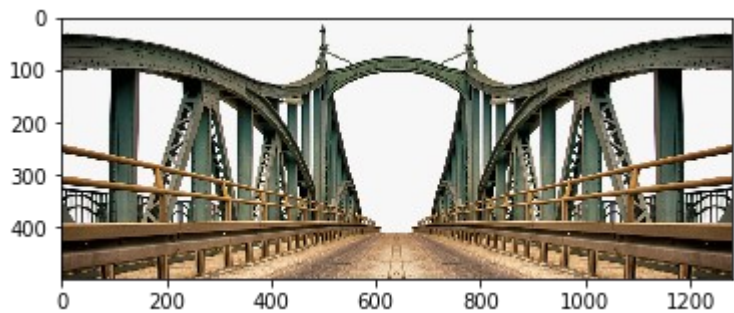


“Corners”



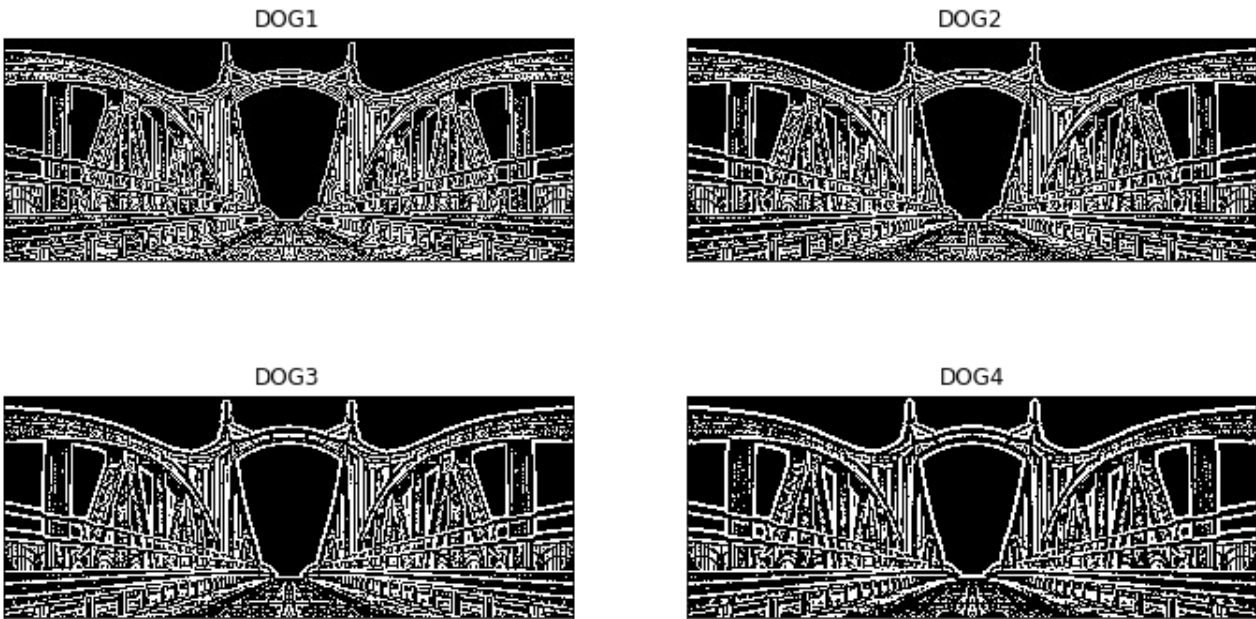
3. Sift Key Points Detection:

Input Image



DOG Images

“Octave 1 with DOG”



“Octave 2 with DOG”



DOG3



DOG4



*“Octave 3 with DOG ”*

DOG1



DOG2



DOG1



DOG2



*“Octave 4 with DOG ”*

DOG1



DOG2



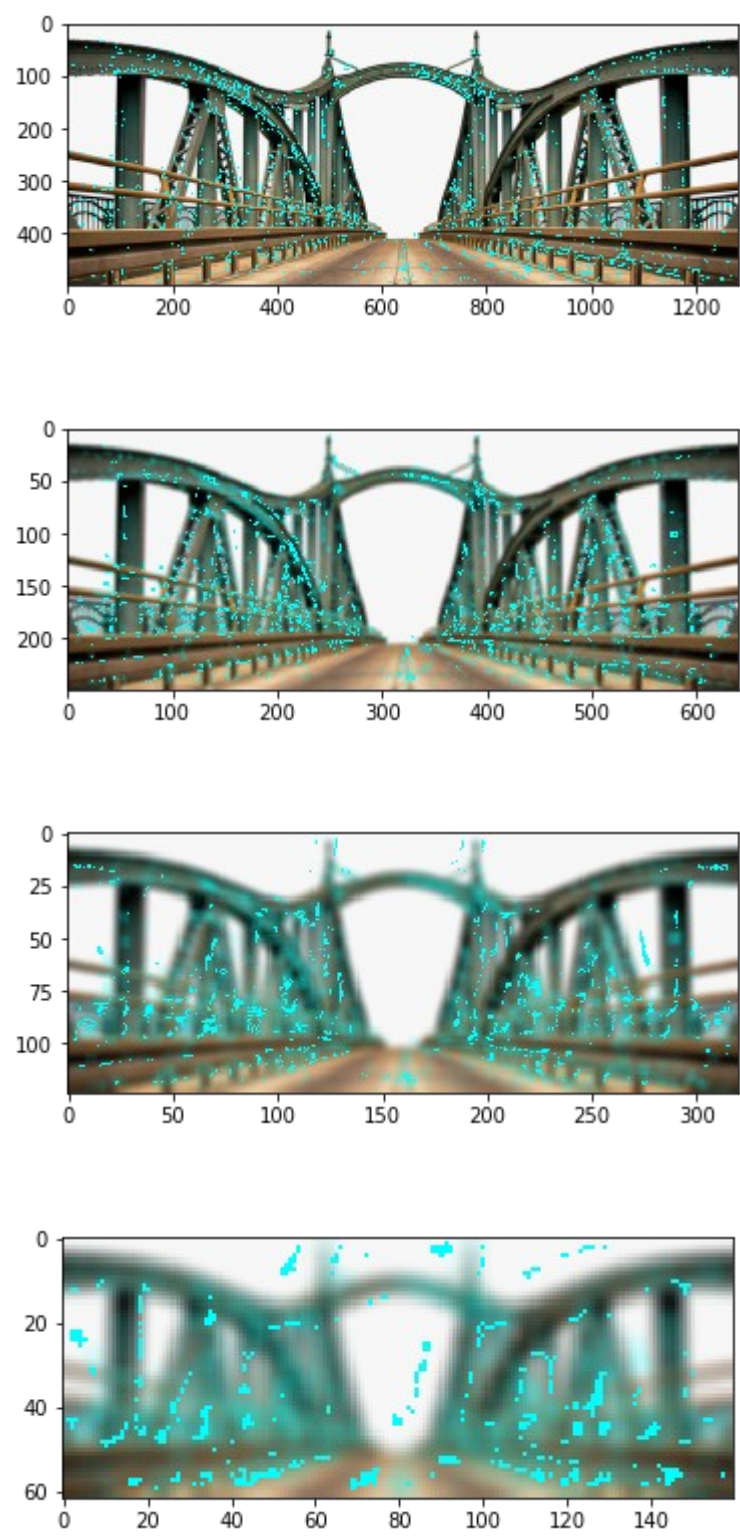
DOG3



DOG4



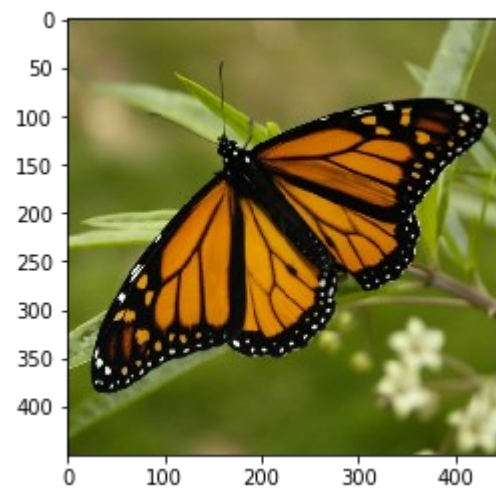
*Output Images*





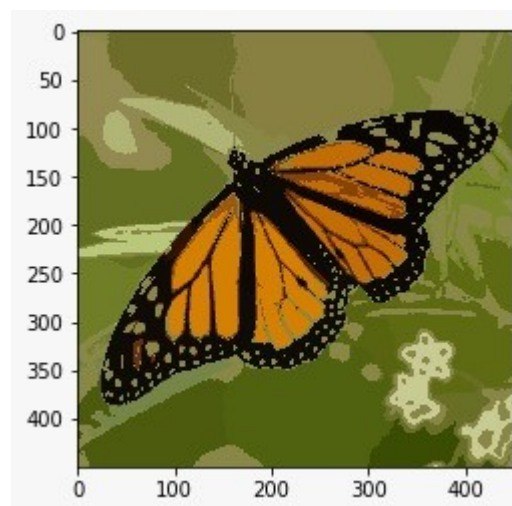
#### 4. Mean Shift

*Input Image:*

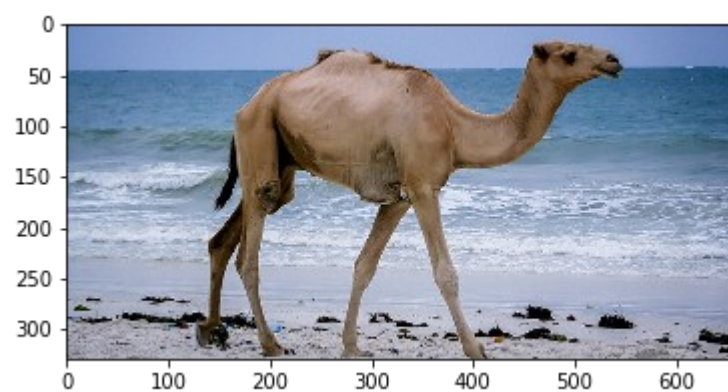


*Ouput Image:*

***For bandwidth = 0.2***

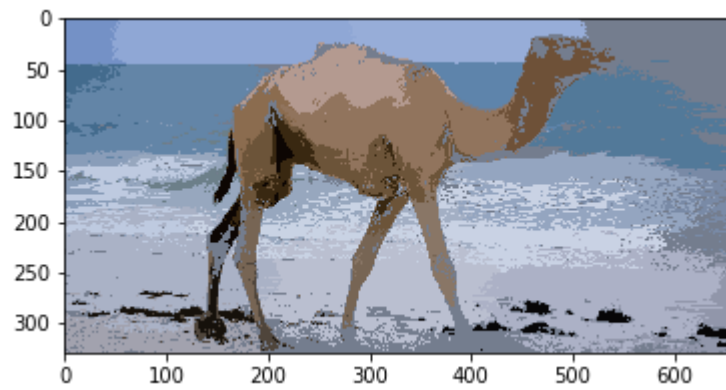


*Second Image*





*Output Image*

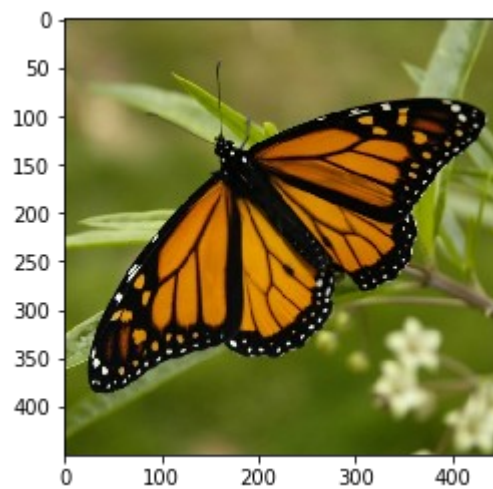


## 5. SLIC

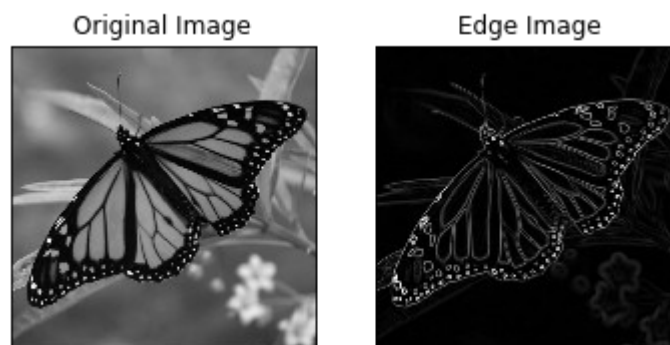
**Formula Used:**

$$D = d_{\text{color}} + (m/S) * d_{\text{space}}$$

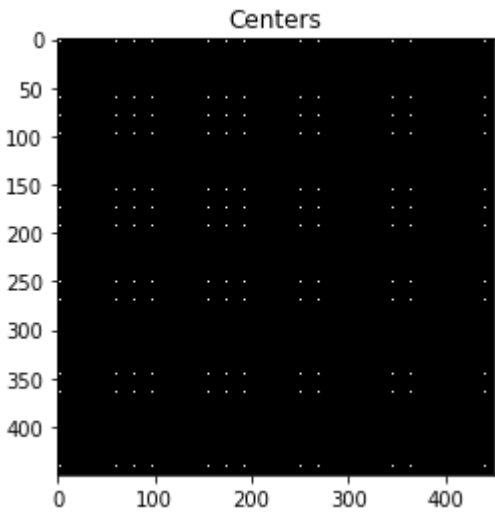
*Input Image*



*Gray and Edge Image:*



*Centres:*



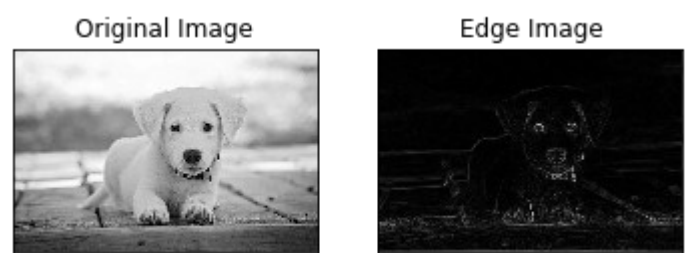
*Output Image:*



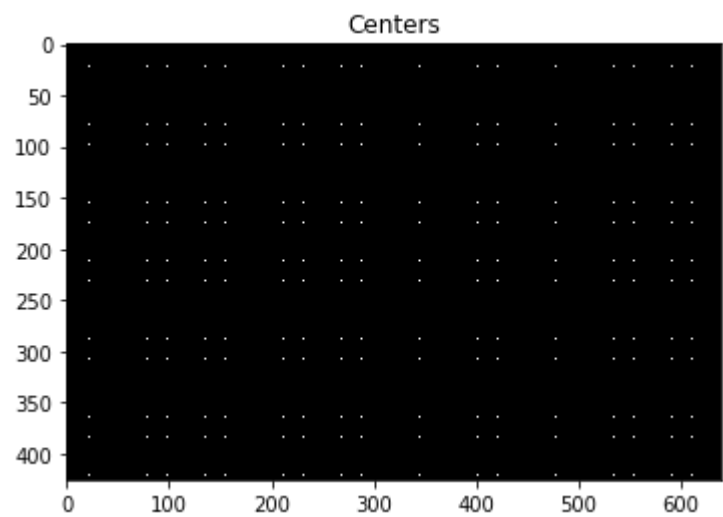
Second Input Image:



Gray and Edge Image:



Centres:



*Output Image*

