**Link :**

https://github.com/devendra-70/CV\_TA\_2

**Algorithms Used :**

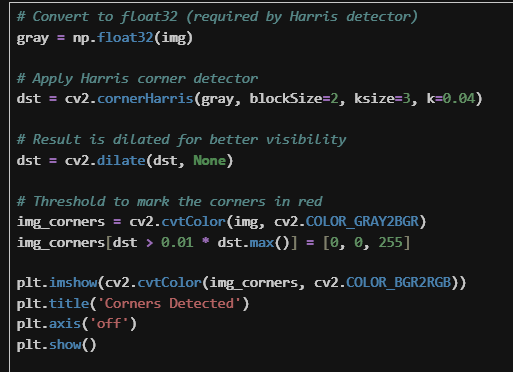
* **Harris Corner Detection**  
  *To detect corners in an image by measuring intensity variation in all directions.*
* **Shi-Tomasi Corner Detection**  
  *To identify the strongest and most reliable corner points using an improved Harris method.*
* **Canny Edge Detection**  
  *To detect sharp edges in an image by finding areas with rapid intensity change.*

**🔹 1. Harris Corner Detection**

**Theory:**  
Detects corners by analyzing intensity variations in a local window.  
Uses the structure tensor and eigenvalues to measure change in gradient directions.  
Corners are points where both directional gradients are strong.

**Parameters:**

* blockSize: Size of the window; larger detects coarser corners.
* ksize: Aperture for Sobel; higher values smooth gradients more.
* k: Sensitivity factor; lower k detects more corners, higher is stricter.

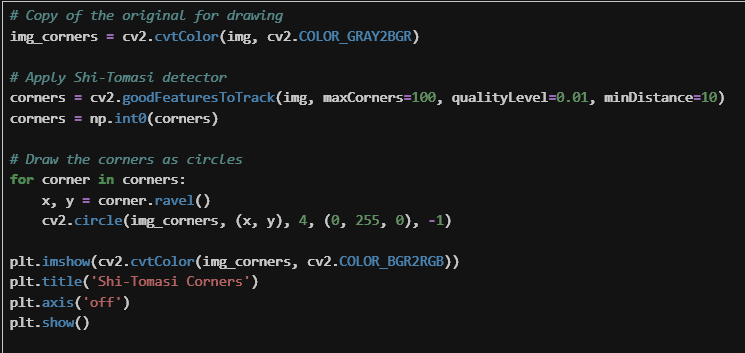


**🔹 2. Shi-Tomasi Corner Detection**

**Theory:**  
Improves Harris by selecting corners based on the **minimum** eigenvalue.  
Focuses on quality and reliability of corner points.  
Ideal for tracking as it avoids unstable corners.

**Parameters:**

* maxCorners: Max number of corners to return; higher = more points.
* qualityLevel: Minimum accepted corner quality (0–1); lower gives more.
* minDistance: Minimum pixel distance between corners; smaller = denser points.



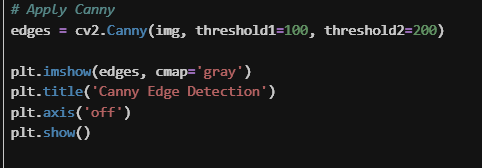


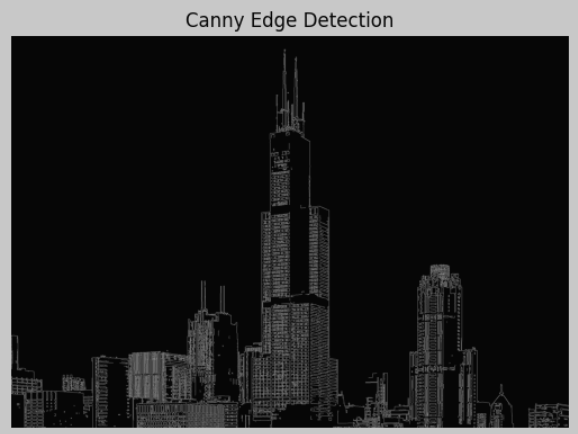
**🔹 3. Canny Edge Detection**

**Theory:**  
Detects edges via gradient intensity, non-maximum suppression, and double thresholding.  
Finds strong and weak edges, and connects them based on continuity.  
Produces clear, thin outlines ideal for object detection.

**Parameters:**

* threshold1: Lower hysteresis threshold; lower = more edges.
* threshold2: Upper hysteresis threshold; higher = cleaner edges.
* *(optional)* Blurring before Canny reduces noise and false edges.





**Github Screenshots:**

