* **Explain why your chosen parameter works better**

**1. GaussianBlur (Kernel size: 11x11):**

**Purpose:** This smooths the image and hence reduces the noise before adaptive thresholding is applied.

**Why 11x11?:** The amount of smoothing applied depends on the size of the kernel. Using a larger (11x11) kernel will smooth the data more, which can be useful if your background varies subtly in intensity. Using a smaller kernel, such as 3x3 in this case, would not remove as much noise, showing more background texture.

**Effect:** Applying the medium-sized kernel (11x11) makes a balance between noise removal and excessive smoothening of the edges of the berry.

**2. Threshold Adaptive (Block size: 11, Constant: 2):**

**Block Size (11):** It is the size of the neighborhood considered around each pixel to compute the threshold.

A smaller block size would consider only pixels in the neighborhood, leading to very detailed, often noisy segmentation. A larger block size, however, for example 11x11, would better handle global variations in illumination. This block size has been chosen to handle the relatively uniform lighting in the image.

**Constant (2):** The amount subtracted from the average of the block to refine thresholding. A small constant like 2 will preserve small changes in brightness, like highlights on the berry, rather than making the entire berry a white blob.

**Effect:** This would dynamically calculate the threshold for the different regions in an image to let the berry standout from its background, even if that background is not uniformly lit.

### **Why These Parameters Work Well:**

* **Balance**: The chosen kernel sizes and the block size for adaptive thresholding were selected to provide a good balance between noise reduction and detail preservation.
* **Customization to Image**: The image has a soft background and subtle lighting variations, which the 11x11 block size and Gaussian blur handle well.