**Bilateral Filtering**

Bilateral smoothing is an edge-preserving denoising technique. As we might have noticed in the other image smoothing techniques, they generally blur out the edges. That does not happen with Bilateral smoothing, and that is why it is called an 'edge-preserving smoothing technique. But how does it preserve the edges?

We already know that in the Gaussian smoothing technique, a weighted sum of all the pixel values in the kernel area is calculated, and the central element of the kernel is replaced with that value. But this is a function of space alone. It is not considered if the pixel lies on edge or not. This is why the Gaussian smoothing technique tends to blur out the boundaries also.

Bilateral filtering or Bilateral smoothing technique overcomes this disadvantage by introducing another Gaussian filter that considers the variation of intensities to preserve the edges.

Bilateral filtering can be implemented in [OpenCV](https://www.projectpro.io/article/opencv-python/792) using the *cv2.bilateralFilter()* function, which takes the following parameters

* src: Image which is to be smoothened
* d: Dimension of the kernel
* sigmaColor: Standard deviation that controls the influence of pixels with different intensity values
* sigmaSpace: Standard deviation that controls the influence of distant pixels

blur = cv2.bilateralFilter(image,9,350,350)

Certainly! The adjustment parameters in bilateral filtering control the behavior of the filter and affect how it considers the neighboring pixels during the filtering process. Here are the key parameters you can adjust:

1. **Diameter (d):**
   * The **d** parameter specifies the diameter of each pixel neighborhood that is used during filtering. It determines how far the filter looks from the central pixel to gather information. A larger diameter includes more pixels in the neighborhood, making the filter more effective at smoothing larger structures but potentially blurring small details.
2. **Sigma Color (sigma\_color):**
   * The **sigma\_color** parameter controls the standard deviation of color space. It determines how different two pixel colors can be while still being considered as neighbors. A larger **sigma\_color** allows a wider range of colors in the neighborhood, which can help preserve edges between regions with different colors.
3. **Sigma Space (sigma\_space):**
   * The **sigma\_space** parameter controls the standard deviation in coordinate space. It determines how far the filter considers neighboring pixels in terms of spatial distance. A larger **sigma\_space** means that pixels farther away from the central pixel have a greater influence on the filtering process. This helps preserve edges and fine details in the image.

Adjusting these parameters involves finding a balance between smoothing and preserving important image features, such as edges. Here are some general guidelines:

* **Smaller Values:**
  + Smaller values of **d**, **sigma\_color**, and **sigma\_space** result in a more local filter. It may not smooth the image effectively or may only smooth small details.
* **Larger Values:**
  + Larger values of **d**, **sigma\_color**, and **sigma\_space** result in a more global filter. It may smooth the image more but could potentially blur edges and fine details.

It's often necessary to experiment with different parameter values based on the characteristics of your image and your specific requirements. You can visually inspect the results and adjust the parameters until you achieve the desired balance between smoothing and edge preservation.