In OpenCV, Gaussian Blur, Erosion, and Dilation are fundamental image processing techniques:

1. Gaussian Blur:
   - Purpose: Smoothens an image by reducing noise and details.
   - How it Works: It uses a Gaussian kernel (a matrix) where the values are calculated using the Gaussian function. This kernel is convolved (slid over) the image to produce a blurred effect.
   - Use in OpenCV: `cv2.GaussianBlur(src, ksize, sigmaX)`, where `src` is the source image, `ksize` is the kernel size, and `sigmaX` is the standard deviation in the X direction.

2. Erosion:
   - Purpose: Erodes away the boundaries of the foreground object (usually white on black).
   - How it Works: It involves a kernel (a small shape or template) that slides through the image (a process known as convolution). A pixel in the original image (either 1 or 0) will be considered 1 only if all the pixels under the kernel are 1, otherwise it is eroded (made to zero).
   - Use in OpenCV: `cv2.erode(src, kernel, iterations)`, where `src` is the input image, `kernel` is the structuring element, and `iterations` is the number of times the operation is applied.

3. Dilation:
   - Purpose: Increases the size of the foreground object.
   - How it Works: It is the opposite of erosion. Here, a pixel element is '1' if at least one pixel under the kernel is '1'. It increases the white region in the image or the size of the foreground object.
   - Use in OpenCV: `cv2.dilate(src, kernel, iterations)`, similar to erosion but it tends to enlarge the boundaries of the foreground object.

These operations are particularly useful in preprocessing steps in image processing and computer vision, such as reducing noise, separating overlapping objects, and even in edge detection. They are fundamental in morphological operations, which involve the shape or structure of features in an image.