

OpenCV Python Image Smoothing – Gaussian Blur

Image Smoothing using OpenCV Gaussian Blur

As in any other signals, images also can contain different types of noise, especially because of the source (camera sensor). Image Smoothing techniques help in reducing the noise. In OpenCV, image smoothing (also called blurring) could be done in many ways. In this tutorial, we shall learn using the Gaussian filter for image smoothing.

Gaussian filters have the properties of having no overshoot to a step function input while minimizing the rise and fall time. In terms of image processing, any sharp edges in images are smoothed while minimizing too much blurring.

Syntax – cv2.GaussianBlur() function

OpenCV provides `cv2.gaussianblur()` function to apply Gaussian Smoothing on the input source image. Following is the syntax of `GaussianBlur()` function :

```
dst = cv2.GaussianBlur(src, ksize, sigmaX[, dst[, sigmaY[, borderType=BORDER_DEFAULT]]] )
```

Parameter	Description
src	input image
dst	output image
ksize	Gaussian Kernel Size. [height width]. height and width should be odd and can have different values. If ksize is set to [0 0], then ksize is computed from sigma values.
sigmaX	Kernel standard deviation along X-axis (horizontal direction).
sigmaY	Kernel standard deviation along Y-axis (vertical direction). If sigmaY=0, then sigmaX value is taken for sigmaY
borderType	Specifies image boundaries while kernel is applied on image borders. Possible values are : <code>cv.BORDER_CONSTANT</code> <code>cv.BORDER_REPLICATE</code> <code>cv.BORDER_REFLECT</code> <code>cv.BORDER_WRAP</code> <code>cv.BORDER_REFLECT_101</code> <code>cv.BORDER_TRANSPARENT</code> <code>cv.BORDER_REFLECT101</code> <code>cv.BORDER_DEFAULT</code> <code>cv.BORDER_ISOLATED</code>

Example – OpenCV Python Gaussian Blur

In this example, we will read an image, and apply Gaussian blur to the image using `cv2.GaussianBlur()` function.

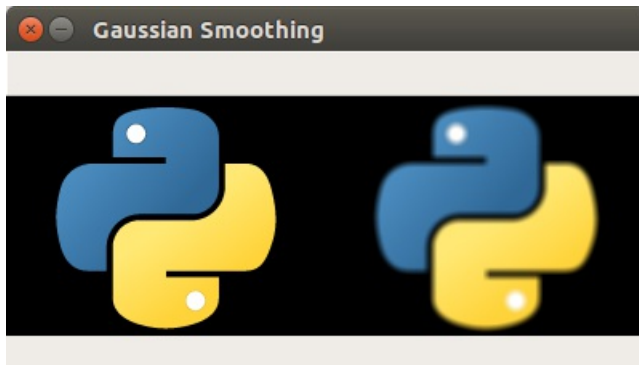
gaussian-blur-example.py

```
import cv2
import numpy

# read image
src = cv2.imread('/home/img/python.png', cv2.IMREAD_UNCHANGED)

# apply gaussian blur on src image
dst = cv2.GaussianBlur(src, (5,5), cv2.BORDER_DEFAULT)

# display input and output image
cv2.imshow("Gaussian Smoothing", numpy.hstack((src, dst)))
cv2.waitKey(0) # waits until a key is pressed
cv2.destroyAllWindows() # destroys the window showing image
```



Now let us increase the Kernel size and observe the result.

```
dst = cv2.GaussianBlur(src, (10,10), cv2.BORDER_DEFAULT)
```



You may change values of other properties and observe the results.

Conclusion

In this [OpenCV Python Tutorial](#), we have learned how to blur or smooth an image using the Gaussian Filter.

OpenCV Python Tutorial

┆ OpenCV Python Tutorial

┆ OpenCV - Setup with Anaconda

┆ OpenCV - Read and Display Image

┆ OpenCV - Save Image

┆ OpenCV - Get Image Shape/Dimensions

┆ OpenCV - Resize Image - Upscale, Downscale

┆ OpenCV - Read Image with Transparency Channel

Image Processing

┆ OpenCV - Edge Detection

┆ OpenCV - Gaussian Blur