



[Data Structures](#) [Algorithms](#) [Interview Preparation](#) [Topic-wise Practice](#) [C++](#) [Java](#) [Python](#)

# Python | Image blurring using OpenCV

Difficulty Level : Medium • Last Updated : 17 Apr, 2019

**Image Blurring** refers to making the image less clear or distinct. It is done with the help of various low pass filter kernels.

## Advantages of blurring:

Attention geek! Strengthen your foundations with the [Python Programming Foundation](#) Course and learn the basics.

To begin with, your interview preparations Enhance your Data Structures concepts with the [Python DS](#) Course. And to begin with your Machine Learning Journey, join the [Machine Learning - Basic Level Course](#)

- It helps in Noise removal. As noise is considered as high pass signal so by the application of low pass filter kernel we restrict noise.
- It helps in smoothing the image.
- Low intensity edges are removed.
- It helps in hiding the details when necessary. For e.g. in many cases police deliberately want to hide the face of the victim, in such cases blurring is required.

## Important types of blurring:

- **Gaussian Blurring:** Gaussian blur is the result of blurring an image by a Gaussian function. It is a widely used effect in graphics software, typically to reduce image noise and reduce detail. It is also used as a preprocessing stage before applying our machine learning or deep learning models.



E.g. of a Gaussian kernel(3×3)

$$\frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

- **Median Blur:** The Median Filter is a non-linear digital filtering technique, often used to remove noise from an image or signal. Median filtering is very widely used in digital image processing because, under certain conditions, it preserves edges while removing noise. It is one of the best algorithms to remove Salt and pepper noise.
- **Bilateral Blur:** A bilateral filter is a non-linear, edge-preserving, and noise-reducing smoothing filter for images. It replaces the intensity of each pixel with a weighted average of intensity values from nearby pixels. This weight can be based on a Gaussian distribution. Thus, sharp edges are preserved while discarding the weak ones.

**Below is the Python code:**

```
# importing libraries
import cv2
import numpy as np

image = cv2.imread('C://Geeksforgeeks//image_processing//fruits.jpg')

cv2.imshow('Original Image', image)
cv2.waitKey(0)

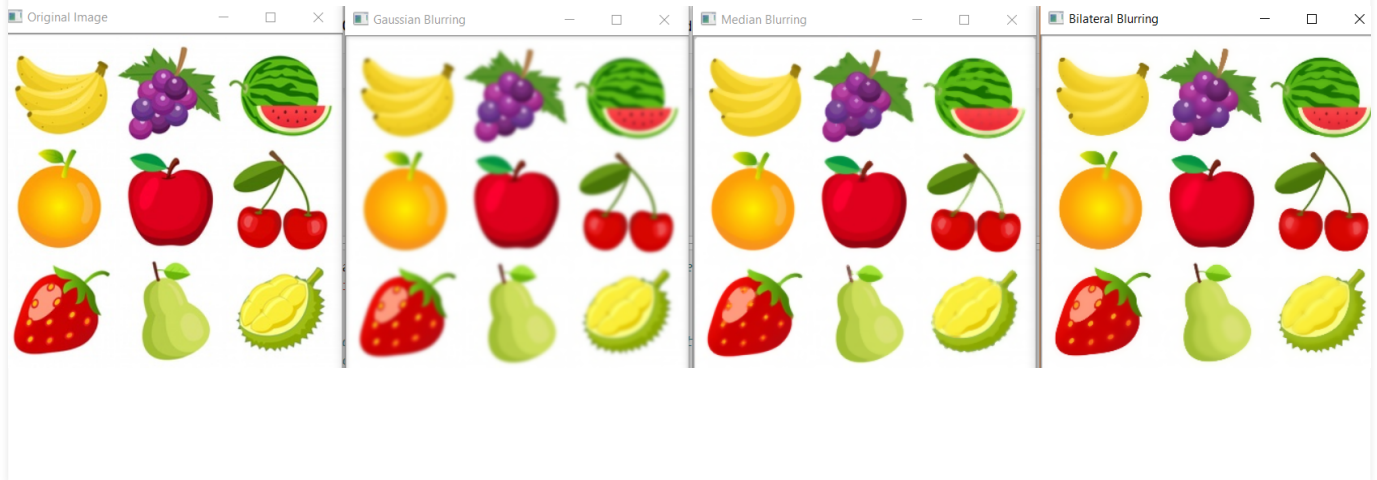
# Gaussian Blur
Gaussian = cv2.GaussianBlur(image, (7, 7), 0)
cv2.imshow('Gaussian Blurring', Gaussian)
cv2.waitKey(0)

# Median Blur
median = cv2.medianBlur(image, 5)
cv2.imshow('Median Blurring', median)
cv2.waitKey(0)

# Bilateral Blur
bilateral = cv2.bilateralFilter(image, 9, 75, 75)
cv2.imshow('Bilateral Blurring', bilateral)
cv2.waitKey(0)
cv2.destroyAllWindows()
```



**Output:**



Like 8

Previous

Next

## RECOMMENDED ARTICLES

Page : 1 2 3

**01** What is Image Blurring  
14, Oct 18

**02** Convert OpenCV image to PIL image  
in Python  
17, Dec 20

**03** OpenCV - Facial Landmarks and  
Face Detection using dlib and  
OpenCV  
18, May 20

**05** Image resizing using Seam carving  
using OpenCV in Python  
17, Aug 20

**06** OpenCV Python Program to analyze  
an image using Histogram  
19, Apr 17

**07** Reading an image in OpenCV using  
Python  
14, Mar 18