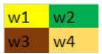
kernel

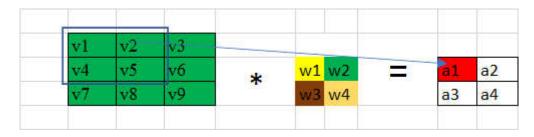
Giả sử kernel là một ma trận có kích thước nxn



Khi ta nhân một ảnh 2D với kernel trên(nhân từng trọng số của kernel với giá trị pixel tương ứng rồi lấy tổng)

v1	v2	v3
v4	v5	v6
v7	v8	v9

khi đó ta được một ma trận mới

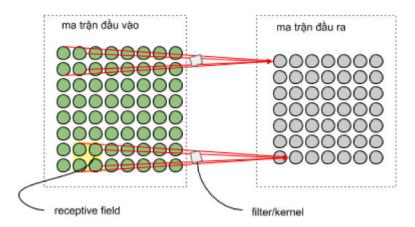


a1 = v1w1 + v2w2 + v4w3 + v5w4

tương tự a2, a3, a4 khi đó ta được ma trận có kích thước nhỏ hơn ban đâu, để giữ lại kích thước ban đâu ta thêm vào các cột, hàng giá trị 0 vào ma trận



Ta có thể tưởng tượng kernel là một cái cửa số quét qua ảnh



MEAN

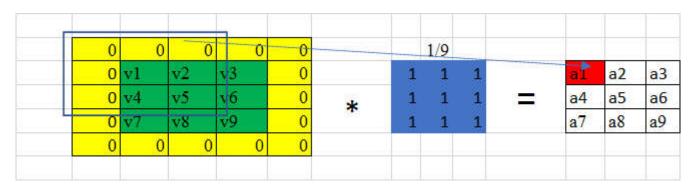
Giá trị trung bình là khái niệm cơ bản trong xác suất thống kê với công thức

$$\overline{X} = \frac{\sum X}{N}$$

Ứng dụng làm mờ ảnh - tức là lấy trung bình giá trị của một vùng pixel khi đó làm giảm chất lượng của ảnh



Ở trên ta đã nói đến kernel, khi đó chỉ thay hết phần tử của kernel bằng một rồi sau đó chia tổng tìm đc cho số phần tử trong kernel thì được giá trị trung bình.



In [7]:

```
import cv2
import numpy as np
# doc dnh xám
img = cv2.imread('image.jpg',0)
print(type(img))# xem kiểu dữ Liệu của ảnh

#tạo kernel
n = 20
kernel = np.ones((n,n), np.float32)/ n**2

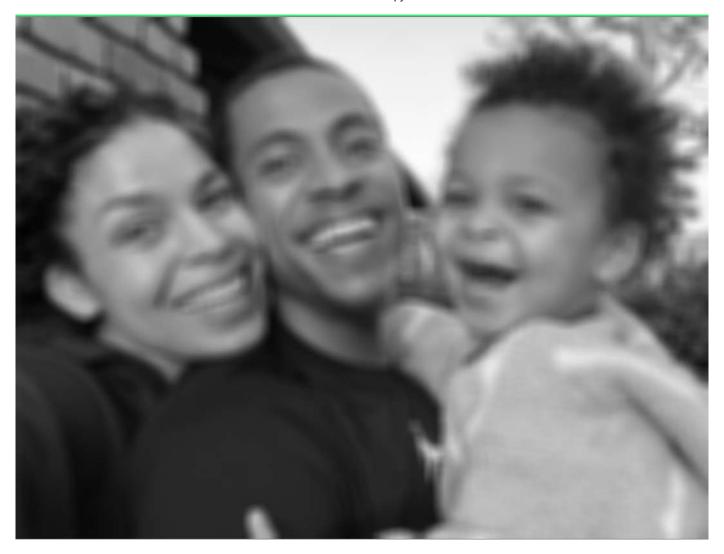
#tính giá trị mean cho từng pixel
mean = cv2.filter2D(img, cv2.CV_8U, kernel)

cv2.imshow("image", img)
cv2.imshow("mean", mean)

cv2.waitKey(0)
cv2.destroyAllWindows()
```

<class 'numpy.ndarray'>





_	-	-
Tn	l	
T11	l	
	_	-