

Traslación de imagen con OpenCV

Importar librerías

```
In [1]: 1 #!/pip install imutils
```

```
In [2]: 1 # Importar los paquetes necesarios  
2 from matplotlib import pyplot as plt  
3 import numpy as np  
4 import imutils  
5 import cv2
```

Función para mostrar imágenes en Jupyter Notebooks y Google Colab

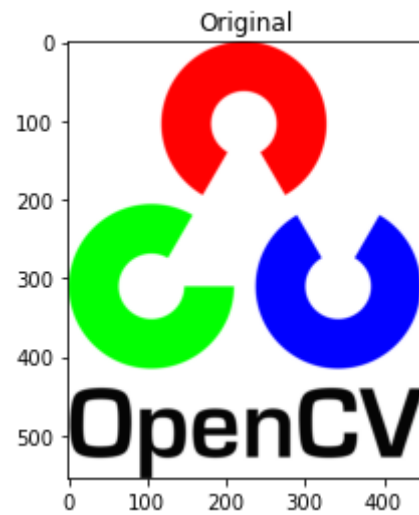
```
In [3]: 1 def mostrar_imagen(titulo,imagen):  
2     imagen = cv2.cvtColor(imagen,cv2.COLOR_BGR2RGB)  
3     plt.imshow(imagen)  
4     plt.title(titulo)  
5     plt.grid(False)  
6     plt.show
```

Image translation with OpenCV

```
In [6]: 1 args = {  
2     "image": "opencv_logo.png"  
3 }
```

In [7]:

```
1 # cargar la imagen y mostrarla
2 image = cv2.imread(args["image"])
3 mostrar_imagen("Original", image)
```

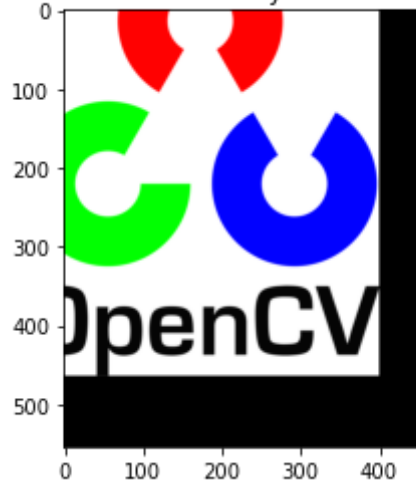


```
In [8]: 1 # trasladar la imagen 25 pixeles a la derecha y 50 pixeles hacia abajo
2 M = np.float32([[1, 0, 25], [0, 1, 50]])
3 shifted = cv2.warpAffine(image, M, (image.shape[1], image.shape[0]))
4 mostrar_imagen("Trasladado hacia Abajo y a la Derecha", shifted)
```



```
In [9]: 1 # ahora, trasladar la imagen 50 pixeles a la izquierda y 90 pixeles
2 # hacia arriba, usando valores negativos para las direcciones x e y
3 # respectivamente
4 M = np.float32([[1, 0, -50], [0, 1, -90]])
5 shifted = cv2.warpAffine(image, M, (image.shape[1], image.shape[0]))
6 mostrar_imagen("Trasladado hacia arriba y hacia la izquierda", shifted)
```

Trasladado hacia arriba y hacia la izquierda



```
In [ ]: 1 # uso de la función de ayuda de imutils para trasladar la imagen 100
2 # pixeles hacia abajo en una simple llamada de función
3 shifted = imutils.translate(image, 0, 100)
4 mostrar_imagen("Trasladado hacia abajo", shifted)
```