
TEMA 6. TRANSFORMACIONES GEOMÉTRICAS

Librerías que tenemos que importar:

- **opencv:** Open Computer Vision (Visión Artificial Abierta), librería para visión artificial de código abierto y que está disponible para muchos lenguajes de programación. Entre ellos se encuentra Python.
 - **numpy:** librería de python especializada en el cálculo numérico y el análisis de datos, especialmente para un gran volumen de datos; da soporte para crear vectores y matrices grandes multidimensionales, junto con una gran colección de funciones matemáticas de alto nivel para operar con ellas.
 - **matplotlib.pyplot:** mostrar gráficos matemáticos e imágenes (como gráficos).
-

Dada la imagen gato.png, realiza las transformaciones abajo indicadas:



Ejercicio 1:

Traslada la imagen 200 píxeles a la derecha y abajo. Busca la función de OpenCV apropiada. Averigua qué argumentos se deben añadir para obtener algo así:



Ejercicio 2:

Realiza ahora una transformación como en la imagen:



Ejercicio 3:

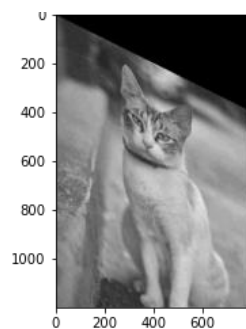
Realiza las operaciones apropiadas para obtener una imagen como esta:



Aparte de hacer la reflexión, necesitarás unir la imagen original con la transformada, para lo que usar una función que *concatene* las matrices.

Ejercicio 4:

Realiza una operación apropiada para obtener una imagen como ésta:

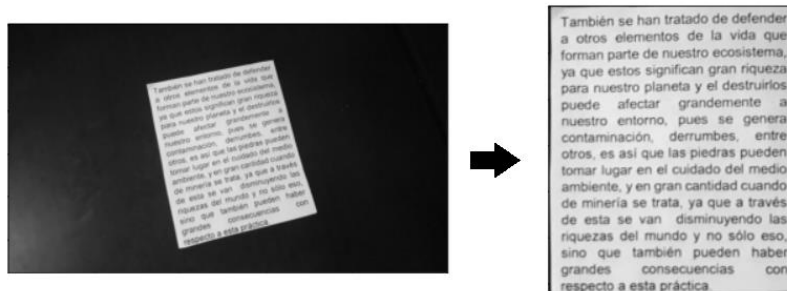


Ejercicio 5:

Recorta en la imagen original la cara del gato y aumenta el tamaño al doble. Usa los argumentos apropiados de la función `resize` de OpenCV para explorar los distintos tipos de interpolación vistos en clase.

Ejercicio 6:

Dada la imagen **documento.jpg**, obtener una imagen en la que se visualice solo el documento centrado, mediante una transformación proyectiva.



Para ello, usa que las coordenadas de los 4 puntos correspondientes a los vértices de la imagen del texto son:

```
pts1 = np.float32([[320,140],[620,85],[395,520],[720,440]])
```