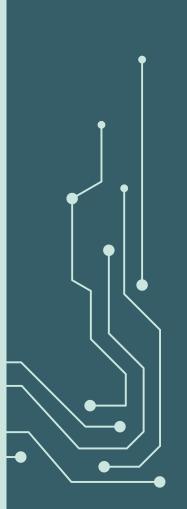


Reconocimiento de texto multicolor

HERRAMIENTAS COMPUTACIONALES: EL ARTE DE LA COMPUTACIÓN

KIMBERLY LUZ VILLET
ROSALIA BEGOÑA GUTIERREZ IZE



PROBLEMA A SOLUCIONAR

En muchas imágenes, el texto aparece en distintos colores y se mezcla con fondos complejos, lo que dificulta su reconocimiento automático mediante OCR. El objetivo es crear un sistema que permita detectar y aislar texto por color específico, facilitando su lectura y extracción automática.

IMPLEMENTACIÓN DE LA RESPUESTA

- QUÉ HACE: LEE EL TEXTO DEL COLOR QUE ELIJAS (ROJO, VERDE, AMARILLO, NEGRO O AZUL).
- CÓMO EMPIEZA: CARGA LA IMAGEN Y LA PASA A RGB.
- DETECTA COLOR: CREA UNA MÁSCARA PARA ESE COLOR Y DEJA LO DEMÁS EN BLANCO.
- LIMPIA PARA LEER: CONVIERTE A GRISES, UNE TRAZOS Y AGRANDA LA IMAGEN.
- LEE EL TEXTO: USA OCR (TESSERACT) Y SE QUEDA CON EL RESULTADO MÁS CLARO.
- REPITE SI QUIERES: MUESTRA EL TEXTO Y TE PREGUNTA SI PROBAR OTRO COLOR.
- TERMINA: CUANDO DICES "NO".

FILTROS

- CONVERSIÓN DE ESPACIO DE COLOR: CV2.CVTCOLOR(IMG, CV2.COLOR_BGR2RGB)
- MÁSCARA DE COLOR: CV2.INRANGE() PARA AISLAR EL COLOR SELECCIONADO.
- ESCALADO Y SUAVIZADO: CV2.RESIZE() CON INTERPOLACIÓN BICÚBICA.
- BINARIZACIÓN: CV2.THRESHOLD() CON VERSIONES NORMAL E INVERTIDA.



ARCHIVO ENTRADA

AMARILLO AZUL NARANJA
NEGRO ROJO VERDE
MORADO AMARILLO ROJO
NARANJA VERDE NEGRO
AZUL ROJO MORADO
VERDE AZUL NARANJA

ARCHIVO SALIDA

```
¿Qué color quieres detectar? (rojo/verde/amarillo/negro/azul): rojo
Texto detectado (rojo):
AZUL MORADO VERDE AZUL
¿Quieres intentar con otro color? (si/no): no
Proceso terminado ☑
```

CONCLUSIONES

- EL MÉTODO PERMITE DETECTAR Y EXTRAER TEXTO POR COLOR, SIENDO ÚTIL EN CONTEXTOS DONDE LOS TEXTOS SE DIFERENCIAN POR RESALTADOS O COLORES (CARTELES, DOCUMENTOS, SEÑALIZACIONES).
- LA EFECTIVIDAD DEPENDE DE:
 - CORRECTA DEFINICIÓN DE RANGOS DE COLOR.
 - CALIDAD DE LA IMAGEN (RESOLUCIÓN, ILUMINACIÓN, CONTRASTE).
- ES UNA SOLUCIÓN ADAPTABLE: PUEDE AMPLIARSE PARA DETECTAR OTRA VARIEDAD DE COLORES.

```
(window).scrollTop() > heade
f (parseInt(header1.css('padd
header1.css('padding-top',
          ader1.css('padding-top', ''
    /indow).scrollTop() > header2_in
(parseInt(header2.css('padding-top', 'interpretation))
header2.css('padding-top', 'interpretation)
            -2 ccs('nadding-top', ''
```