

Algoritmo 7

Creación de máscaras de imágenes usando el modelo HSV

En este algoritmo veremos cómo construir una máscara de una imagen de color para aplicar el cambio de fondo de una imagen.

Introducción

Supongamos que tenemos una imagen con fondo verde como la siguiente



Nuestro objetivo es eliminar el fondo verde para tener algo como esta imagen

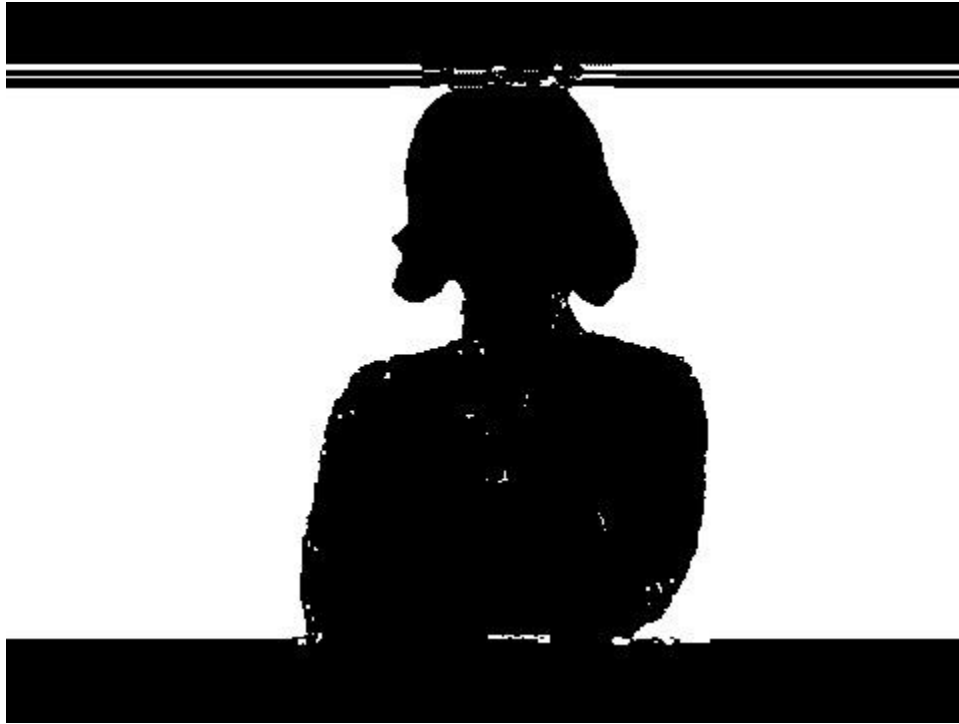


Para obtener este resultado necesitamos de varios pasos, así que vamos a sepáralos para que sea más claro cómo realizarlo.

Objetivo

El objetivo de este algoritmo es encontrar los pixels del fondo verde.

En el caso de la imagen anterior, el fondo de la imagen se representa con el color blanco y lo que no con color negro, es decir como en la siguiente imagen:



Es importante tomar en cuenta que es una primera aproximación, por lo que aparecen algunos pixels blancos en la silueta.

Creación de máscaras de imágenes

Para construir las máscaras de la imagen utilizaremos la función `cv2.inRange`, la cual tiene la siguiente sintaxis

```
im2 = cv2.inRange(im, valorMenor, valorMayor)
```

En donde:

- `Im` es la imagen de entrada, en nuestro caso en formato `hsv`
- `valorMenor` es una tupla de 3 valores, con los valores mínimos de Hue, Saturation y Value
- `valorMayor` es una tupla de 3 valores, con los valores máximos de Hue, Saturation y Value
- `im2` es la imagen resultado

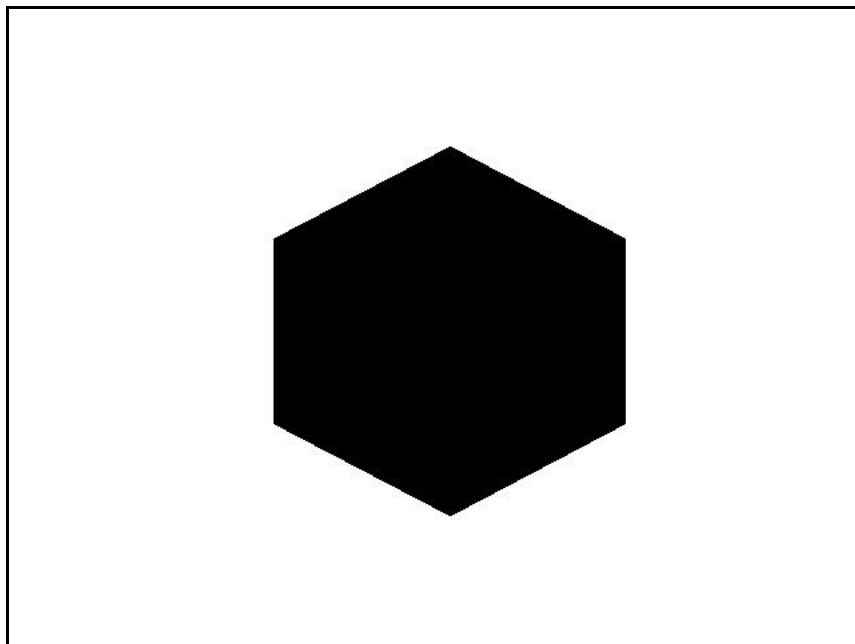
En otras palabras, la función `cv2.inRange` nos regresa en color blanco todos los pixels de la imagen que se encuentren entre los valores mínimo y máximo. El resultado de la función es una imagen a la que llamamos máscara.

Ejemplo

Supongamos que tenemos la siguiente imagen



Y queremos identificar los pixels que son de color verde para obtener la siguiente imagen



Nota que todos los pixels del fondo son ahora de color blanco.

Para hacer lo anterior, podemos usar el siguiente código

```
import cv2
```

```
im = cv2.imread('ejemplo.png')
```

```
hsv= cv2.cvtColor(im, cv2.COLOR_BGR2HSV)
```

```
mask= cv2.inRange(hsv,(69,0,0),(69,255,255))
```

```
cv2.imshow('imagen', mask)
```

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

Ejercicio 1

Construye una máscara para cada color fondo de las imágenes:
ejercicio1.png, ejercicio2.png, ejercicio3.png, ejercicio4.png y ejercicio5.png

En este caso la máscara de la imagen debería ser la misma ya que son las mismas figuras, sin embargo, los valores son diferentes ya que el color del fondo es diferente.

Ejercicio 2

1. Busca tres imágenes con fondo verde (green screen) en internet
2. Busca los mejores valores para encontrar los pixels del fondo de cada imagen y construir la máscara de la imagen
3. Muestra tus resultados

Código

Incluye tu código donde se calculan y muestran las máscaras de las imágenes.

Reporte

En tu reporte deberás mostrar todos los resultados obtenidos, incluye los valores utilizados para generar las máscaras de la imagen.