

Asignatura	Datos del alumno	Fecha
Percepción Computacional	Apellidos: Romero Perez	27/02/2023
	Nombre: Juan carlos	

Laboratorio: Evaluación de la segmentación

Introducción

A lo largo de los años los seres humanos hemos desarrollado diferentes herramientas conforme a los avances científicos, una de las más importantes se deriva de los dispositivos con cámaras e incluso equipos médicos como lo son las radiografías, los cuales nos permiten capturar un momento y guardarlo para ser utilizado en un futuro con un fin en específico.

Uno de los procesos más importantes que realizan las cámaras, es el de segmentación, ya que nos permite separar e identificar objetos de una imagen o video, eliminando las cosas que no son de relevancia para el análisis o proceso que se requiera realizar la segmentación en la imagen.

En el presente trabajo se utilizan 2 algoritmos para segmentar 2 imágenes y se realiza una comparativa para determinar en qué situación un algoritmo funciona mejor que otro.

Asignatura	Datos del alumno	Fecha
Percepción Computacional	Apellidos: Romero Perez	27/02/2023
	Nombre: Juan carlos	

Desarrollo

Algoritmo 1

Una de las técnicas de segmentación más utilizadas, debido a que es una función similar a la visión en el ser humano (la detección de colores), es la segmentación por color, para realizar el presente trabajo se propuso el primer algoritmo, el cual se basa en las técnicas de segmentación por color HSV.

Para entender mejor cómo funcionan los canales HSV en la librería OpenCV, el autor Gastón Di Giuseppe nos proporciona la siguiente información de utilidad:

“En OpenCV, los canales de HSV varían así: [H:0–179, S:0–255, V:0–255]. Debemos entender la variación de H como “grados”, que en realidad varía de 0 a 360, como vemos en la imagen, pero en OpenCV lo simplifican en 180 grados. Vemos también que la saturación (S) varía desde el centro hacia afuera y el valor (V) desde lo más oscuro, la base, hasta lo más claro, el techo.

“

Entendiendo los canales HSV, se propuso un algoritmo en Python, se implementó una clase en la cual estarán todos los métodos necesarios para realizar la correcta segmentación de los objetos, dentro de la clase ColorHSV se encuentran los siguientes métodos:

- `convertirARGB(self, img)`: Este método sirve para convertir la imagen a una escala de grises.
- `convertirAHSV(self, img)`: Se le pasa la imagen original y se convierte a un formato HSV.
- `binRange(self, h, hmin, hmax)`: Para este método se realiza un proceso de obtención de rangos de los colores, como se observa en la figura 1, el rango para la imagen del gato fue de 5,30 y para el kiwi se aplicó un rango de 31,44. Una vez proporcionado los rangos, el método realizara un proceso de binarización de la imagen, tal y como se observa en la figura 2.

Asignatura	Datos del alumno	Fecha
Percepción Computacional	Apellidos: Romero Perez	27/02/2023
	Nombre: Juan carlos	

Figura 1

Obtención de los rangos de matiz

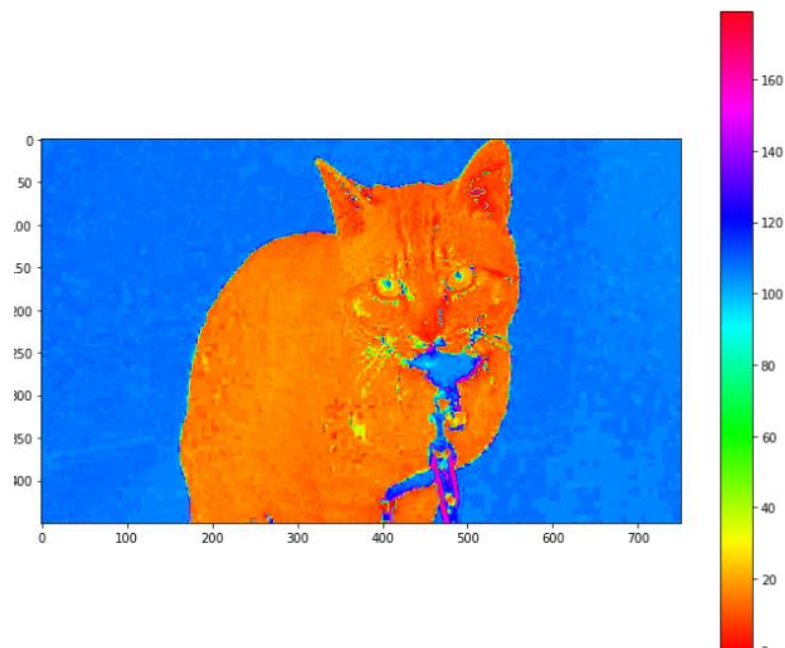
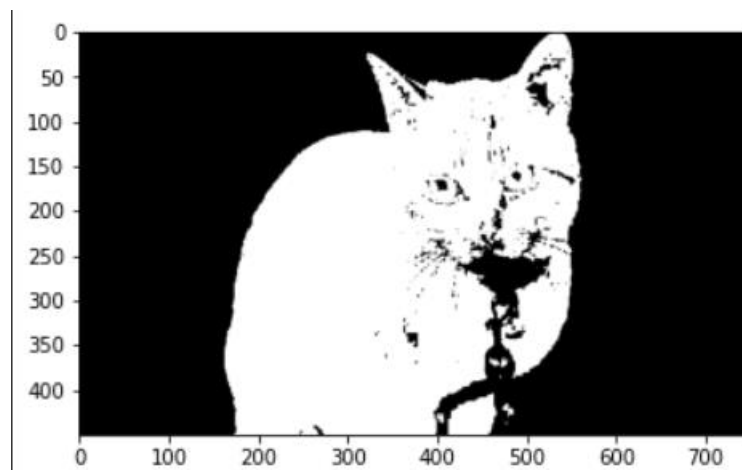


Figura 2

Imagen después de aplicarle la función `binRange(self,h,hmin,hmax)`

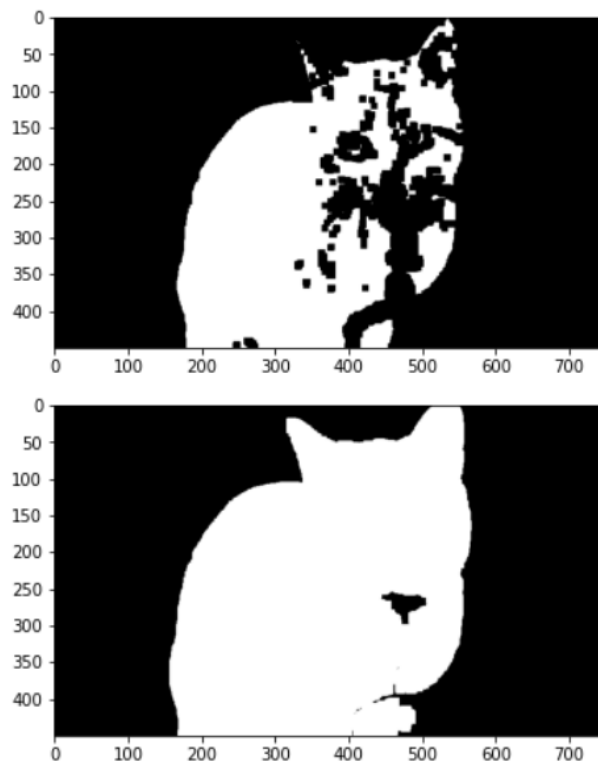


- `filtradoMor(self,binimg)`: Con este método se realiza un proceso de apertura morfológica para eliminar el ruido de la imagen en zonas no deseadas.

Asignatura	Datos del alumno	Fecha
Percepción Computacional	Apellidos: Romero Perez	27/02/2023
	Nombre: Juan carlos	

Figura 3

Imagen final después de aplicar el filtrado de apertura



Algoritmo 2

Este algoritmo se basa en la segmentación por colores RGB, el cual se maneja con 3 colores principales el rojo, verde y azul. Este espacio está basado en un sistema cartesiano de coordenadas x,y.

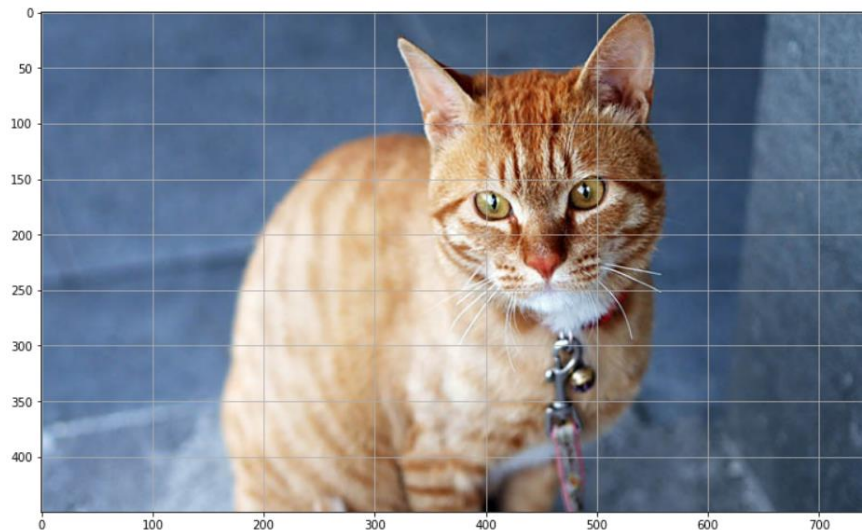
Para este algoritmo se propuso la clase llamada ColorRGB, la cual está constituida por los siguientes métodos:

- `convertirARGB(self, img)`: Este método sirve para convertir la imagen a una escala de grises.
- `binRange(self,img,h,l,h2,l2)`: Mediante esta función se extraerá se elige un rango para extraer el conjunto de pixeles mediante el cual se va segmentar por color. El rango se elige utilizando una foto con las coordenadas a elegir, en el caso de la foto del gato, se tomó las coordenadas (200,300,200,300) .

Asignatura	Datos del alumno	Fecha
Percepción Computacional	Apellidos: Romero Perez	27/02/2023
	Nombre: Juan carlos	

figura 4

Ejemplo de elección del rango mediante coordenadas



Los siguientes métodos se definieron para establecer los valores mínimos y máximos para cada uno de los colores rojo, verde y azul de la zona extraída, es por esto que a estas funciones se les pasa las coordenadas de la proporción elegida de la fotografía.

- `rangeRmin(self,rl)`: extracción de la región de interés rango mínimo del color rojo
- `rangeRmax(self,rl)`: extracción de la región de interés rango máximo del color rojo.
- `rangeGmin(self,rl)`: extracción de la región de interés rango mínimo del color verde.
- `rangeGmax(self,rl)`: extracción de la región de interés rango máximo del color verde.
- `rangeBmin(self,rl)`: extracción de la región de interés rango mínimo del color azul.
- `rangeBmax(self,rl)`: extracción de la región de interés rango máximo del color azul.

Asignatura	Datos del alumno	Fecha
Percepción Computacional	Apellidos: Romero Perez	27/02/2023
	Nombre: Juan carlos	

- `uBajo(self,rmin,gmin,bmin)`: Mediante este método, se obtiene el umbralizado bajo y para esto le pasamos los valores de los rangos mínimos de los 3 colores RGB.
- `uAlto(self,rmax,gmax,bmax)`: Mediante este método, se obtiene el umbralizado alto y para esto le pasamos los valores de los rangos máximos de los 3 colores RGB.
- `binarizacion(self,imgRGB,umbral_bajo,umbral_alto)`: Función para realizar el proceso de binarizacion de la imagen, se le pasan los umbrales altos y bajos obtenidos previamente.
- `filtradoMor(self,binimg)`: Con este método se realiza un proceso de apertura morfológica para eliminar el ruido de la imagen en zonas no deseadas.
-

Resultados

Se probaron los algoritmos con 2 imágenes diferentes, las cuales se muestran a continuación:

Imagen 1

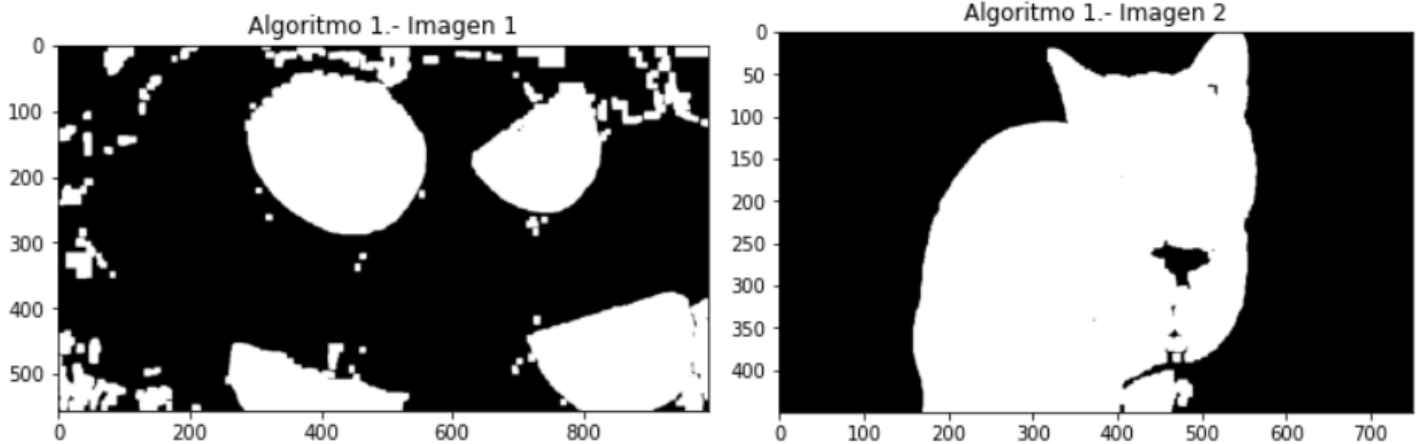


Imagen 2

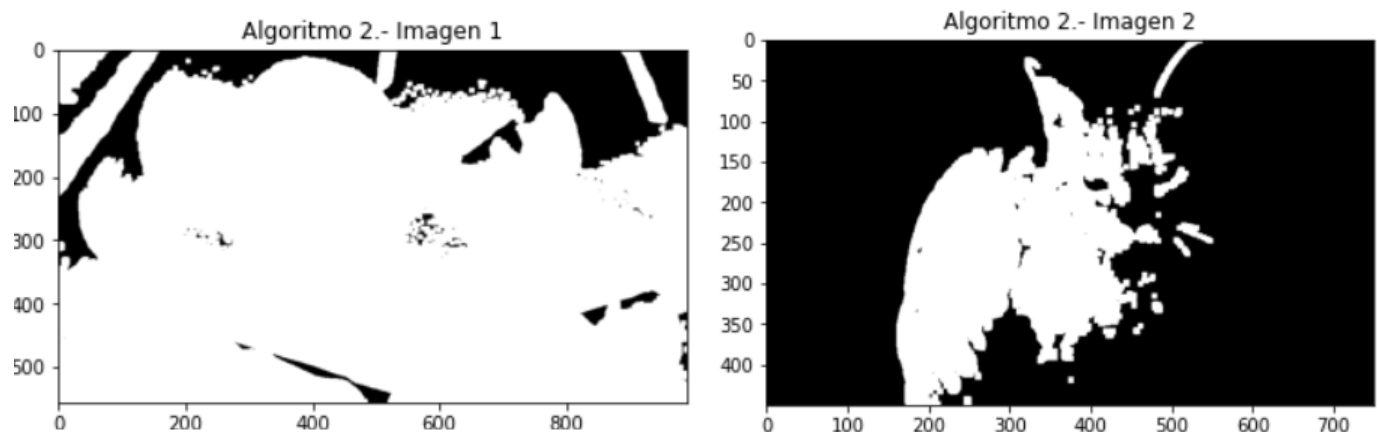


Asignatura	Datos del alumno	Fecha
Percepción Computacional	Apellidos: Romero Perez	27/02/2023
	Nombre: Juan carlos	

Para el algoritmo 1, segmentación por Color CSV los resultados fueron bastantes buenos tal y como se muestran en las siguientes imágenes.



Con el algoritmo 2 se obtuvieron los siguientes resultados en las imágenes.



Para evaluar que tan bueno fue el algoritmo, se utilizó una función llamada jaccard. La cual sirve para medir el grado de similitud entre 2 conjuntos, se mide en porcentaje y un 1 vendría siendo que está perfectamente igual a la imagen ground truth. En la siguiente tabla se muestran los resultados de ambos algoritmos.

	Imagen 1	Imagen 2
Algoritmo 1 (HSV)	0.67728	0.93305
Algoritmo 2 (RGB)	0.25069	0.55292

Asignatura	Datos del alumno	Fecha
Percepción Computacional	Apellidos: Romero Perez	27/02/2023
	Nombre: Juan carlos	

Conclusiones

Analizando los resultados de ambos algoritmos, se llega a la conclusión de que el algoritmo 1 es mejor a la hora de hacer la segmentación, sin embargo me di cuenta que trabaja mejor cuando el fondo es de un solo color y creo que esa es la mayor dificultad de los algoritmos de segmentación, poder filtrar todo el fondo y dejar solo la parte que nos interesa es una tarea difícil. El algoritmo 2 presento mucho margen de error al momento de realizar segmentaciones, sin embargo es probable que para algunas otras imágenes en condiciones adecuadas funcione mejor este algoritmo.

En un futuro me adentrare mas en el procesamiento de imágenes y probare algoritmos como el watershed o algún otro para comprobar su precisión contra estos 2 algoritmos por color.

Como extra se creo una función para aplicar la imagen original ya segmentada a color, pero por temas de espacio no se añadió al reporte, no obstante, en el notebook se puede observar la aplicación del filtro.

Bibliografía

Giuseppe, G.D (2019). *Detección de objetos por colores en imágenes con Python y OpenCV* Recuperado el 20 de 02 de 2023, de <https://medium.com/@gastonace1/detecci%C3%B3n-de-objetos-por-colores-en-im%C3%A1genes-con-python-y-opencv-c8d9b6768ff>

Laguna. [@_Laguna] (24 de abril del 2021). 09 Visión Artificial con Python Segmentación HSV [Archivo de video]. Youtube. <https://www.youtube.com/watch?v=zTrWPbNOCN8&t=931s>

Laguna. [@_Laguna] (24 de abril del 2021). 08 Visión Artificial con Python Segmentación RGB [Archivo de video]. Youtube. https://www.youtube.com/watch?v=TnkBw-LT_FY&t=652s