Instituto Politécnico Nacional

Escuela Superior de Cómputo

Ingeniería en Inteligencia Artificial

Visión artificial

Examen del segundo parcial

Alumno:

Rodrigo Olarte Astudillo

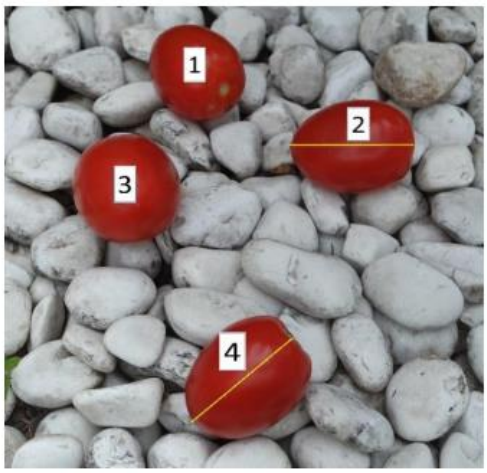
Fecha: 9 de noviembre de 2022

Para la elaboración de este examen se no solicitó realizar lo siguiente:

a. Del algoritmo que se proponga para segmentar los objetos se deben mostrar las imágenes resultado de cada proceso.

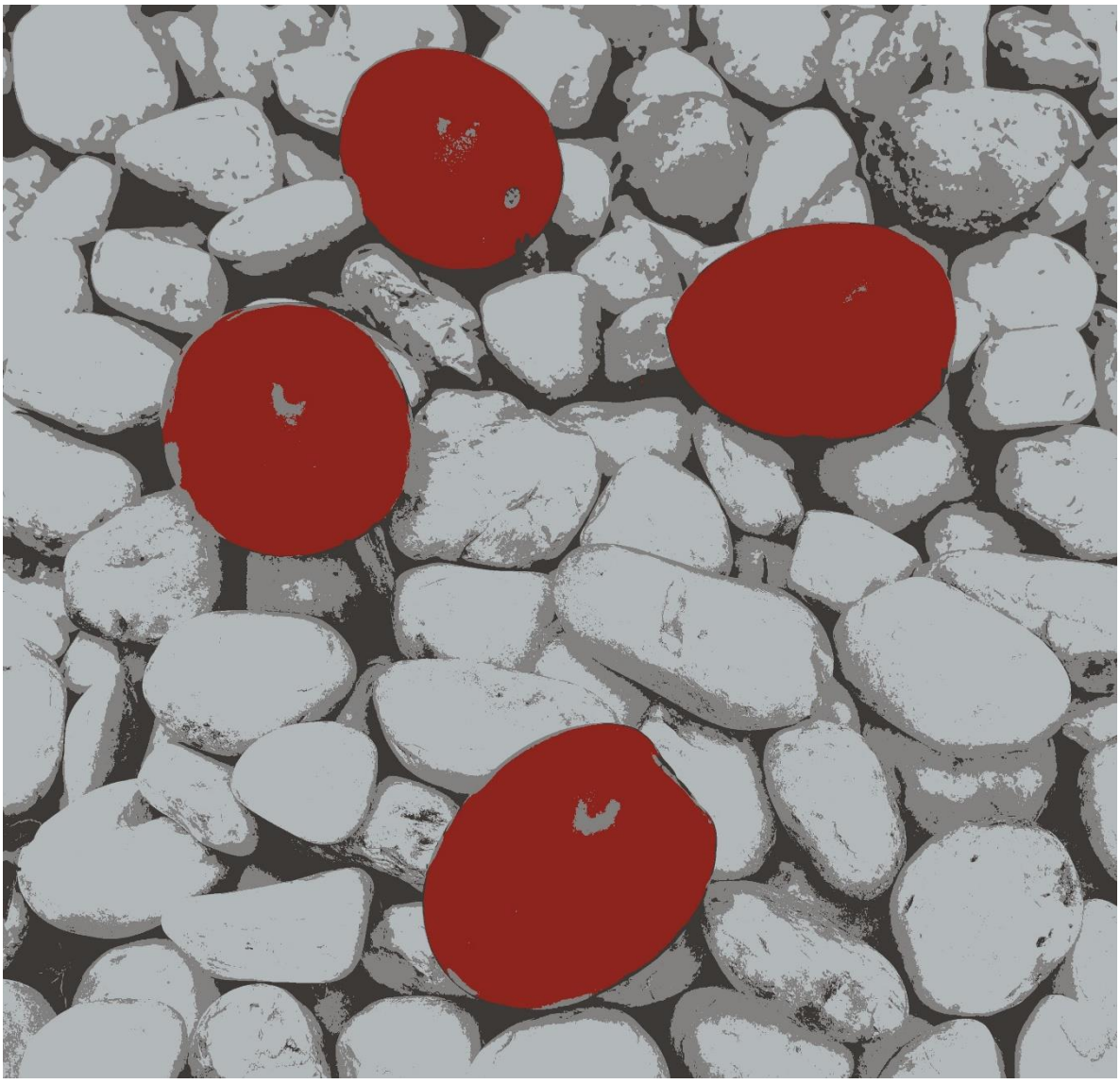
b. Imprimir el valor de las coordenadas encontradas como puntos de medición, recalcando que estas deben ser obtenidas como resultado del proceso automático de segmentación.

c. En la imagen original marcar una línea entre los dos puntos de medición e imprima el valor de la longitud.

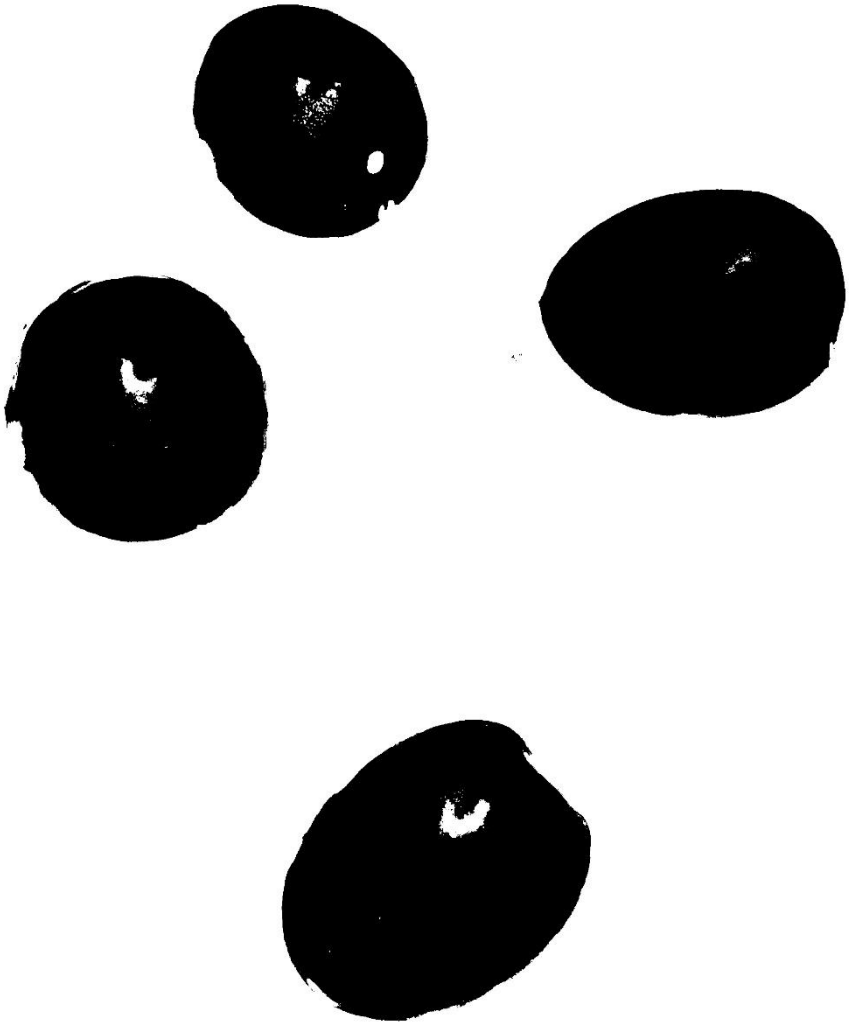


Como primer paso lo que se tiene que hacer es segmentar la imagen, por lo tanto la primera idea era dividir la imagen en 3 matrices dependiendo de R,G,B para aplicar un filtro gauss y así eliminar las imperfecciones, sin embargo, este procedimiento tarda demasiado, por lo tanto se usa k-medios en la imagen original.

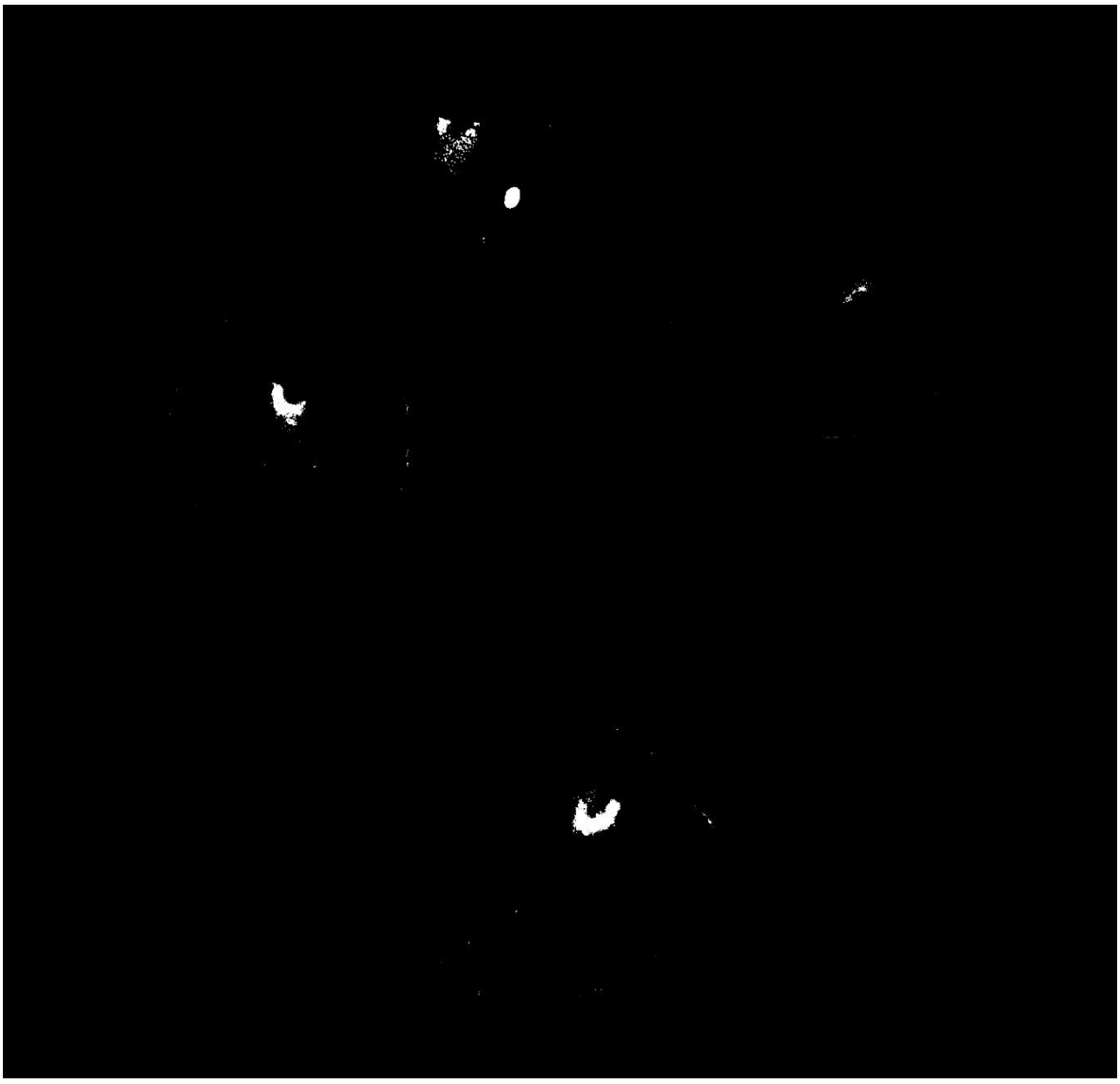
Queda da la siguiente forma:



Una vez que se obtiene la imagen de K-medios, se convierte a binario utilizando la escala de gris de su matriz B, esto ya que los valores del rojo serían casi oscuras, por lo que es relativamente sencillo encontrar sus formas sin contar el fondo o las piedras.



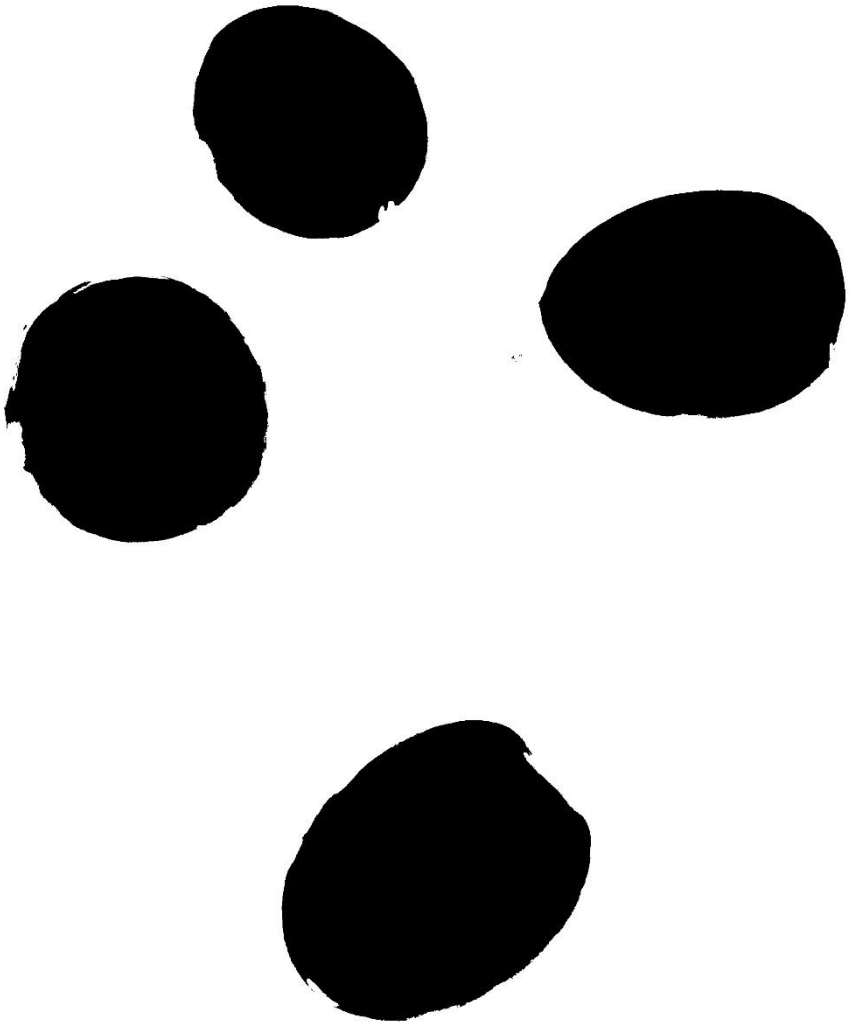
La imagen anterior es la binaria de la k-medios en su escala de grises en el componente azul, dicho esto, podemos ver que tiene imperfecciones en sus centros, esto se debe por el reflejo de la luz en los tomates, por lo tanto, para limpiar esto lo primero que se debe hacer es pintar únicamente el fondo de negro.



Y como se puede apreciar, el reflejo es lo único que nos queda, por lo tanto, el siguiente paso es invertir los valores de esa imagen, es decir, si es blanco se cambia a negro y viceversa.

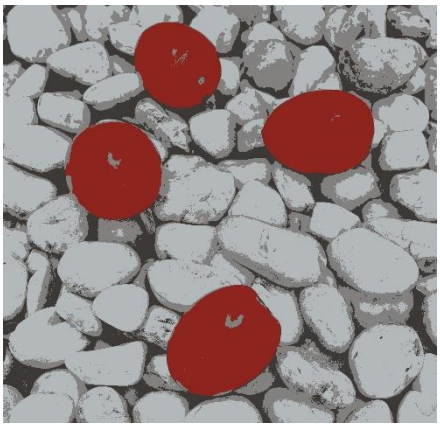
Y como ya se pueden dar cuenta, el siguiente paso es unir esa imagen donde el reflejo es negro con nuestra imagen binaria original.

Y si las unimos nos quedaría de la siguiente forma:

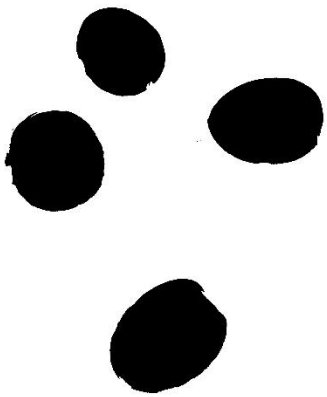


Ahora podemos ver la segmentación que se nos solicitó en el inciso A

Segmentación por color:



Segmentación a binario:



Por último, para el inciso B y C se debe obtener el contorno de los tomates, aquí debo aclarar que no lo guardé en una imagen debido a que me tomaba demasiado tiempo, por lo que en su lugar únicamente guardé sus pixeles en una lista de listas, estos pixeles los detectará como bordes si y solo si sus adyacentes son 255 o blanco, como se puede ver en la siguiente línea de código.

```
#si son bordes o no
for i in range(filas):
    for j in range(columnas):
        adyacentes = False
        adyacentes = (imagen[i-1,j-1]==255 or imagen[i-1,j]==255 or imagen[i,j-1]==255 or imagen[i+1,j+1]==255 or imagen[i+1,j]==255 or imagen[i,j+1]==255 or imagen[i-1,j+1]==255 or imagen[i+1,j-1]==255)
        for index,x in enumerate(colores):
            if (imagen[i,j]==x) :
                lista_clusters_completos[index].append((i,j))
                if (adyacentes):
                    #print(imagen[i,j])
                    #print((imagen[i-1,j-1]==255 or imagen[i-1,j]==255 or imagen[i,j-1]==255 or imagen[i+1,j+1]==255 or imagen[i+1,j]==255 or imagen[i,j+1]==255 or imagen[i-1,j+1]==255 or imagen[i+1,j-1]==
                    #print(adyacentes)
                    #print(imagen[i-1,j-1]==255,imagen[i-1,j]==255,imagen[i,j-1]==255,imagen[i+1,j+1]==255,imagen[i+1,j]==255,imagen[i,j+1]==255,imagen[i-1,j+1]==255,imagen[i+1,j-1]==255,"->",i,j)
                    lista_clusters[index].append((i,j))
                    #imagen[i,j]=0
```

Como se puede apreciar en la imagen anterior, checo sus adyacentes y si alguno cuenta con blanco entonces ese lo guardo como borde, cabe mencionar que no se usó el método de canny o LoG debido a que tardaban demasiado tiempo en ser procesarlos, por lo que no era para nada óptimo con una imagen de 2000x2000 pixeles.

Y con esos bordes uso la fórmula de:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Entre todos los pixeles de una misma lista.

Para saber que pixel corresponde a cada lista solo era necesario basarnos de sus colores, se recorrería de izquierda a derecha y arriba abajo, si se encuentra un valor negro entonces lo cambiaría a un valor de gris de 50+n, donde n=0. Y se cambiarán todos los pixeles que estén conectados con ese píxel donde fue detectado, después si encuentra otro valor negro entonces a n le suma 1 y se le asigna el valor de 50+n = 51.

Así sucesivamente hasta que pasara por toda la matriz y por ende, al finalizar cada tomate tendría un color distinto de 1 pixel de diferencia, por lo tanto, esos colores nos servirían como etiquetas para localizar a todos los pixeles del mismo tomate.

Una vez que ya tenemos todas las distancias y obtengamos las máximas junto con sus puntos, se aplica un producto cruz para encontrar la línea y esto se realiza únicamente en los tomates 2 y 4.

Y la imagen final del examen quedaría de la siguiente forma:

Como se puede apreciar las líneas amarillas y verdes son la distancia y la longitud junto con los puntos de esa imagen son: