

Entrega Guia 1

Introduccion

Mendoza, Agustín - Zacarias, Valentín

1 Ejercicio 1.3

1.1 Análisis

Dada una fotografía en escala de grises se pide identificar la o las botellas que no se encuentren llenas. Para esto se han planteado dos cuestiones centrales: primero contar de una referencia de una botella llena, para poder tener datos certeros los cuales serán utilizados para comparar con las demas; y segundo identificar cada una de las botellas y así realizar su analisis de contenido.

La imagen cuenta con un fondo bien diferenciado de los objetos, por lo que genera que haya un buen contraste con los mismos. Además, el contenido de la botella con el envase también ayuda al procesamiento por el hecho de que tienen un rango de intensidades bien separadas. En cuanto a las dimensiones de las botellas, a simple vista (y como debería de esperarse) todas son del mismo tamaño y estan equidistantes unas de las otras. Si bien algunas no se encuentran completas en la imagen, el alcance de esta implementación esta dado para imagenes en las cuales se encuentre encuadrado el 50% de la botella como minimo, en comparacion a la que se tiene como referencia.

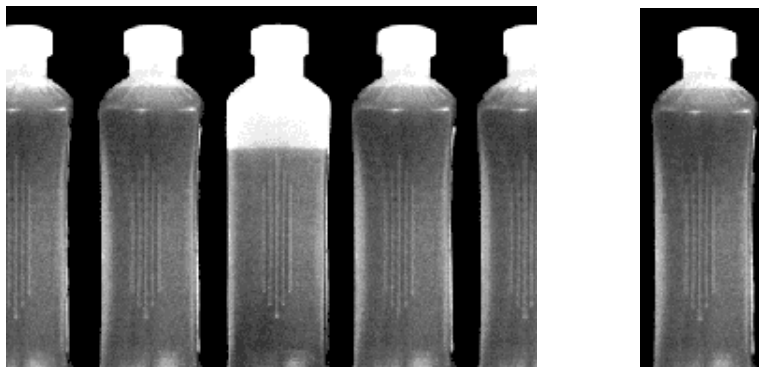


Figure 1: *Imagen a procesar y referencia de botella llena*

1.2 Implementacion

Como primer paso se tomó de referencia una de las botellas llena y completa de la imagen a procesar, recortandola mediante un software de edición. A ésta se le realizó un pre procesamiento mediante una transformación a través del operador intervalo umbral (opUmbralGris), manteniendo la escala de grises en el rango 0-210, mientras que lo que se encuentre por encima es transformado a la intensidad 255. De esta manera se eliminó el ruido que había en las partes donde la botella no tiene contenido.

Luego se procedió a buscar el centro aproximado (en la coordenada x) de la botella. Para esto se implementó la

función `centroBotella`, la cual recorre por filas, pixel a pixel buscando la primer ocurrencia de un pixel blanco (255), el cual indicará el comienzo de la tapa. Encontrada la primer ocurrencia, se guarda la posición de la fila que se está recorriendo y se cuenta la cantidad de pixeles blancos siguientes que hay en esta. Al recorrer las columnas se almacena la posición del ultimo pixel blanco encontrado, que utilizará como identificador para el fin de la tapa, una vez recorrida toda la fila. Con estos datos la función devuelve el punto central de la tapa en coordenadas absolutas a la imagen que recibe como parámetro.

Obtenido este dato se realiza un promedio de la escala de intensidad de la columna en la que se encuentra el centro de la tapa, hacia abajo. Este promedio, con un margen de error a definir, es la referencia que se utilizará para evaluar si cada botella esta llena o le falta contenido. A continuación se pasó a la imagen de interés; en ella se realizó el mismo pre procesamiento. Para realizar el "aislado" y poder evaluar cada botella de forma individual se procedió generando una región de interés. La función `coord_bounding_box` (cuadro inferior) recibe la imagen pre procesada, y recorriendo columna a columna, busca el primer pixel de una intensidad distinta a la del fondo (con intensidad >10) y luego de hallarla, busca la primer ocurrencia de intensidad igual al fondo (<10). Con esta búsqueda queda registrado el inicio y final del objeto en la coordenada y. De manera análoga se procede en la coordenada x. El resultado de esta función es un par de puntos (x,y) opuestos que encierra, de forma aproximada, al objeto. Además se agregó un margen, expandiendo la región para darle robustez al método. El mismo devuelve las coordenadas de la región ante la primer ocurrencia. En esta región delimitada procedemos al igual que con la botella de referencia, obtenemos el centro relativo a la región delimitada, promediamos la intensidad en esa columna y comparamos con la de referencia. En caso de tener un promedio mayor (incluido el error) significa que la botella tiene menos contenido (más cantidad de pixeles blancos). Debido a que la función `centroBotella` devuelve coordenadas absolutas de la imagen de parametro y en este ciclo se esta aislando las botellas por regiones de interés, mantenemos en una variable los datos necesarios para poder obtener las coordenadas absolutas en la imagen original posteriormente, quedando guardadas junto al porcentaje de llenado correspondiente. Luego se descarta la botella analizada, realizando una copia de la imagen recortada, en la que no se cuenta con esta y el ciclo vuelve a comenzar. Una vez recorrida toda la imagen, se imprime en pantalla la imagen original con formato RGB, donde se encuadra (a partir de las coordenadas absolutas que fueron guardadas previamente) en color rojo mediante un bounding box las botellas con faltante de contenido.

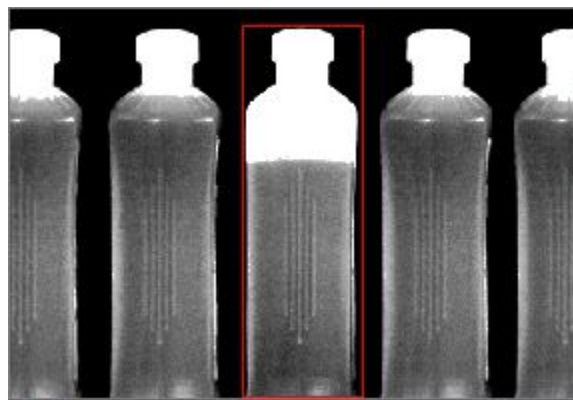


Figure 2: Resultado

```

1 def coord_bounding_box(img, color, margen):
2     x_i = 0
3     y_i = 0
4     x_f = img.shape[1]
5     y_f = 0
6     flag = True
7     #Busco inicio y final a lo ancho (x)
8     for j in range(img.shape[1]):
9         if(flag and np.average(img[:,j]) > 10):
10             x_i = (j-margen if (j-margen)>0 else 0)
11             flag = False
12         if(not(flag) and np.average(img[:,j]) < 10):
13             x_f = (j+margen if (j+margen)<img.shape[1] else img.shape[1])
14             break
15
16     #Busco inicio y final a lo largo (y)
17     flag = True
18     for i in range(img.shape[0]):
19         if(flag and np.average(img[i,:]) > 10):
20             y_i = (i-margen if (i-margen)>0 else 0)
21             flag = False
22         if(not(flag) and np.average(img[i,:]) < 10):
23             y_f = (i+margen if (i+margen)>img.shape[0] else img.shape[0])
24             break
25     y_f = i
26
27     return [(x_i, y_i), (x_f, y_f)]

```

Listing 1: *BoundingBox*

1.3 Conclusiones

Con este trabajo de aplicacion lo que podemos ver es que con conocimientos basicos de lo que es una imagen, sumado a tecnicas de edicion y una matematica sencilla podemos resolver problemas de distincion de elementos similares.

[Si bien el progama se ajusta al alcance pedido en el ejercicio] Por mas de que el programa realizado no tiene la capacidad de generalizacion como para reconocer diferencias en otros objetos que no sean el tipo de botellas especificas que se encuentran en la imagen, si comparamos el resultado obtenido con la poca complejidad de la solucion y su escasa exigencia de poder de computo, vemos que la ganancia es mucha. Pensando en utilidades, éste sistema podria servir para una cadena de produccion de bebidas para identificar productos con menos contenido que el deseado, requiriendo unicamente de una camara que capture las imagenes de las botellas.

Algunas mejoras que se podrian implementar para elevar la utilidad y el desempeño del sistema pueden ser, el trabajar con imagenes a color para asi poder detectar el buen estado del contenido, considerar otros tipos de envases, utilizar como entrada una camara en tiempo real, entre otras. Lo interesante de esto es que si bien estas mejoras implican complejizar la solucion, el nucleo del sistema podria seguir manteniendose sin muchas modificaciones, ya que mientras obtengamos un recorte de la botella y podamos conseguir su centro vertical, el sistema seguira siendo capaz de constatar cual es el grado de capacidad ocupada por el liquido de la botella.