

Eliminación de fondo (Background removal) con ImageJ

Juan Carlos Miranda¹

<http://otroblogdetecnologias.blogspot.com/>,
<https://github.com/freelanceparaguay/>,
juancarlosmiranda81@gmail.com

1 Introducción

La eliminación de fondo (en inglés background removal), es una operación esencial en el tratamiento de imágenes y extracción de características en sistemas de visión de computadoras.

Habitualmente, esta operación es parte de un algoritmo general para la clasificación de objetos y extracción de características a partir de una imagen de entrada [6].

En este artículo presenta en forma práctica la aplicación de operaciones morfológicas y de eliminación de fondo en imágenes de naranjas obtenidas aplicadas a imágenes de naranjas obtenidas desde una base de datos de imágenes pública [1].

Se implementó desarrollando la solución con código Java, utilizando ImageJ [3] como librería de procesamiento de imágenes y Netbeans IDE 8.1 [4] como entorno de desarrollo.

2 ¿Para qué sirve la eliminación de fondo?

¿Para qué sirve la eliminación de fondo?, en primera instancia, permite aislar regiones que se desean analizar, calcular áreas de píxeles o formas geométricas en imágenes.

A continuación algunos ejemplos:

Cálculo de área en píxeles . Es posible asociar unidades de medida de distancia con píxeles mediante una escala. Un caso es obtener el calibre de un objeto.

Extracción de color en zona en una región a analizar . La extracción hace más fácil el análisis de uno o varios colores en una zona elegida. Un ejemplo sería conocer el color predominante en una fruta para su clasificación.

Asociación de varias regiones en una imagen de entrada . Es posible, a partir de una imagen, obtener varios objetos y compararlos con patrones almacenados. En una fotografía con varios tornillos de diferentes tamaños, una tarea sería seleccionarlos, rotarlos y compararlos contra una base de datos.

Un documento interesante que explica la eliminación de fondo es [6], el mismo aplica el proceso para la clasificación de bananas.

3 Algoritmo e implementación

El código fuente de la solución en lenguaje Java puede ser descargado desde el enlace [5].

Se tomaron imágenes de naranjas en **buen estado y en estado de descomposición** desde una fuente de datos pública [1].

Se agruparon las operaciones de procesamiento de imágenes en una clase con las secuencias. Este proceso fue denominado "**Secuencias Morfológicas**".

Como ayuda visual para la prueba de la aplicación de las operaciones se utilizó Fiji ("Fiji Is Just ImageJ") [2], el cual cuenta con un grabador de macros para facilitar la conversión a lenguaje Java de las operaciones aplicadas.

Por último, se procedió a copiar los píxeles desde la imagen a color original, utilizando como máscara los datos obtenidos en la imagen binaria. Estas operaciones se agruparon en un proceso llamado "**Copiado de Imagen**".

Las imágenes elegidas cuentan con un **contraste alto**, de manera a facilitar la extracción de la fruta. Se obtuvieron buenos resultados con imágenes de fondo oscuro Fig.1 y de fondo blanco Fig.2, Fig.3, no obstante también se probó con una imagen cuyo fondo no era uniforme y los resultados no fueron óptimos comparados con las imágenes anteriores, Fig.4 y Fig.5.

Las imágenes utilizadas, sirven a modo pedagógico, en la elección de las mismas no se tomaron en cuenta ángulos de enfoque y métodos relacionados a la adquisición uniforme de imágenes, aunque se recomienda que este proceso responda a una metodología estandarizada.

Fig. 1. Naranja ideal, con fondo negro



Fig. 2. Naranja con moho, con fondo blanco



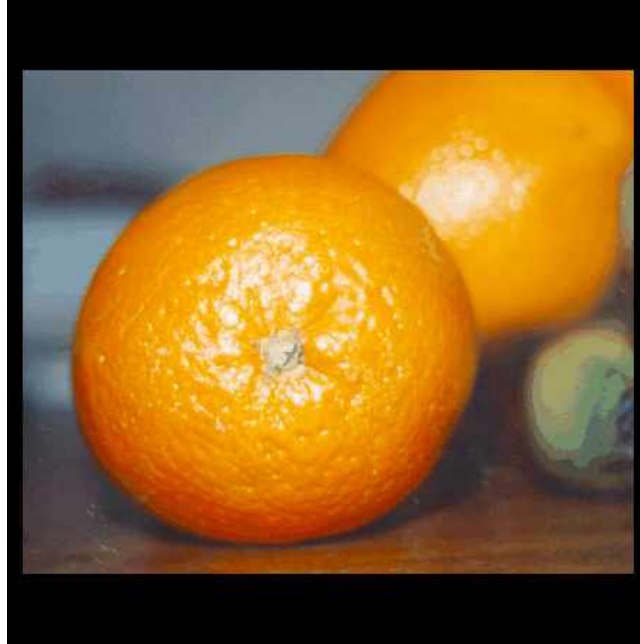
Fig. 3. Naranjas en descomposición, con fondo blanco y sombra



Fig. 4. Naranja sobre madera, el fondo dificulta la extracción uniforme de la región



Fig. 5. Naranjas con fondo variado, se dificulta la extracción del fondo



En el proceso "**Secuencia Morfológica**", se aplicaron las operaciones:

1. Resaltado de bordes.
2. Binarización.
3. Dilatación.
4. Relleno de silueta.
5. Cerradura.

Para el relleno de la silueta, se realizó un barrido del área de la fruta utilizando dos variables apuntadoras para las posiciones de las columnas. Donde el valor 255 representa al color BLANCO y el valor 0 al color NEGRO. Básicamente lo que se hace es establecer valores de tope en una fila y rellenar con un valor correspondiente al color blanco.

El resultado obtenido es una imagen binaria con la silueta, la cual fue utilizada como entrada para el siguiente proceso "**Copiado de Imagen**". Los resultados obtenidos pueden visualizarse en las figuras Fig.6, Fig.7, Fig.8, Fig.9, Fig.10.

Fig. 6. Resultados procesando la imagen Fig. 1



Fig. 7. Resultados procesando la imagen Fig. ??



Fig. 8. Resultados procesando la imagen Fig. 3

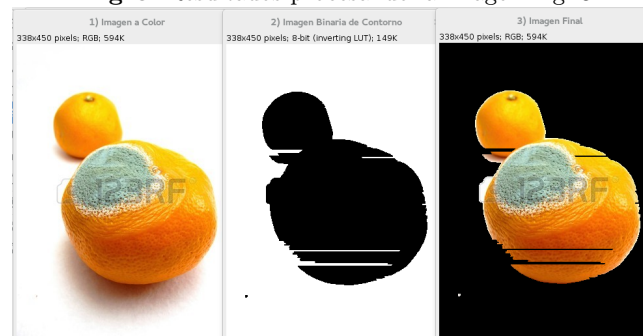


Fig. 9. Resultados procesando la imagen Fig. 4

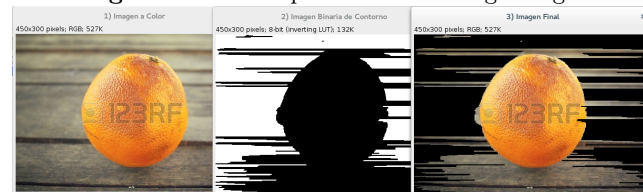
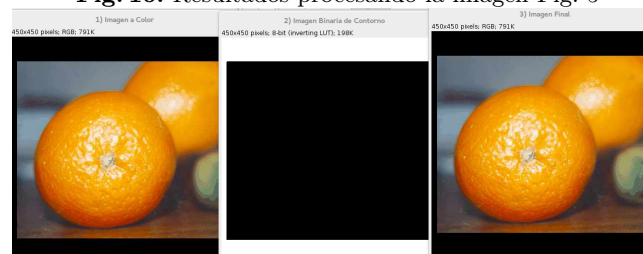


Fig. 10. Resultados procesando la imagen Fig. 5



4 Conclusión

Luego de las pruebas realizadas sobre las imágenes de naranjas, es recomendable utilizar imágenes con un fondo que contraste con la fruta sobre la cual se desea trabajar para eliminar el fondo.

References

1. Banco de imágenes: Fotos libres de derechos y vectores. <http://es.123rf.com/>. [Web; accedido el 26-10-2016].

2. Fiji. <http://imagej.net/Fiji>. [Web; accedido el 28-09-2016].
3. Imagej. <http://imagej.net/ImageJ>. [Web; accedido el 28-09-2016].
4. Netbeans ide. <http://netbeans.org/>. [Web; accedido el 26-10-2016].
5. J. C. Miranda. Proyecto para netbeans separacionfondo. <https://github.com/freelanceparaguay/javaImageJ/tree/master/SeparacionFondo>. [Web; accedido el 26-10-2016].
6. D. Surya Prabha and J. Satheesh Kumar. Assessment of banana fruit maturity by image processing technique. *Journal of Food Science and Technology*, 52(3):1316–1327, 2015.