

# Segmentación de imágenes de un objeto conocido

Daniel Acuña U.

**Abstract**—Segmentar un objeto de interés es un problema común en el procesamiento de imágenes. Este proceso se dificulta cuando la imagen contiene ruido y bajo contraste. En este artículo se presenta un método de segmentación por colores, considerando ruido gaussiano en la imagen.

**Index Terms**—Segmentation, Edge-preserving smoothing, In-painting, Image Processing.

## I. INTRODUCCIÓN

La segmentación de objetos en imágenes requiere de una adecuación previa de la imagen, a fin de facilitar el trabajo del algoritmo principal de segmentación. Considerando un ruido gaussiano en la imagen, se aplica el filtro no lineal Kuwahara, que elimina eficazmente el ruido Gaussiano sin perder los bordes de la imagen como ocurre con el filtro Gaussiano tradicional. Luego podemos hacer otras iteraciones de este mismo filtro para emparejar colores conservando los bordes. De este modo, el filtro que elimina ruido también es parte de la segmentación misma de la imagen. Una vez adaptada la imagen, se crea una máscara del tipo umbral para los colores del objeto de interés.

## II. IMPLEMENTACIÓN ALGORÍTMICA

### A. Filtro Kuwahara

Existen muchas variaciones de este filtro, aquí se muestra una de las más sencillas [1]. Consiste en tomar una ventana en la vecindad de un pixel. Ésta a su vez se subdivide en cuatro segmentos. A cada región se le calcula la varianza y promedio de color. El valor del pixel es reemplazado por el promedio de la región con la menor varianza.

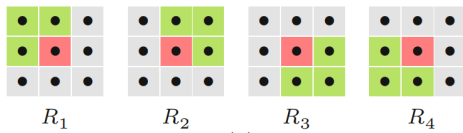


Fig. 1. Subdivisiones de la ventana

La implementación del filtro en imágenes a colores difiere según la definición de varianza de una región. En esta oportunidad utilizaremos la siguiente definición:

1) *Suma de varianzas*: La varianza es definida como la suma de las varianzas por canal en la región.

$$\sigma^2 = \sigma_R^2 + \sigma_G^2 + \sigma_B^2 \quad (1)$$

El filtro considera una máscara cuadrada móvil de 11 pixeles de largo. Se itera este algoritmo sobre el resultado anterior 5 veces para que los colores de cada región queden lo suficientemente regularizados.

### B. Máscara de segmentación

La máscara de segmentación se crea considerando los colores del objeto de interés, que se seleccionan manualmente según sea el caso. Se crea una máscara por umbralización, de modo que el valor de la máscara es cero para colores cercanos a los ideales y uno para colores muy distintos a los ideales. Finalmente, se multiplica la máscara de segmentación con la imagen original sin ruido.

## III. RESULTADOS

En la figura 2 se aprecian los sub-resultados de este filtro.

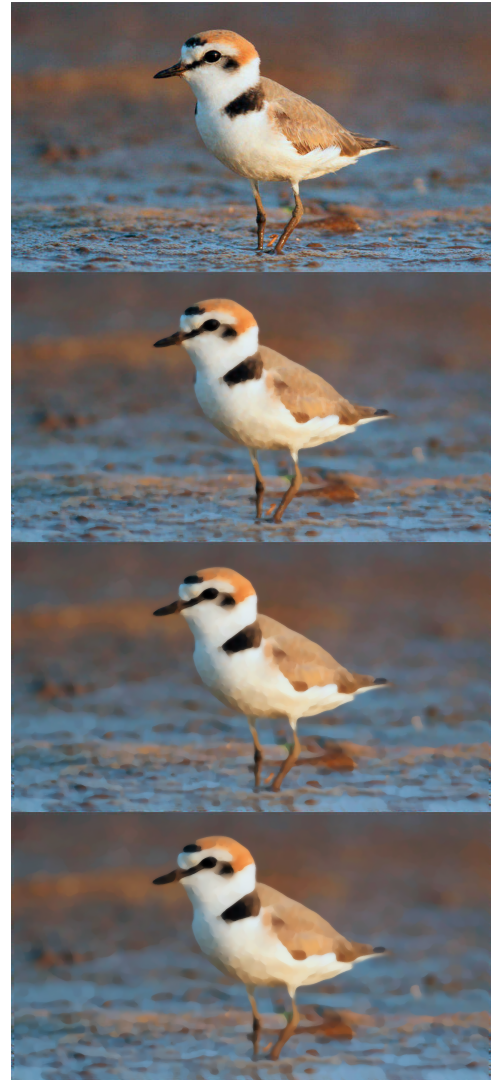


Fig. 2. Iteraciones de filtrado.

En la figura 3 podemos apreciar la máscara obtenida y el resultado.

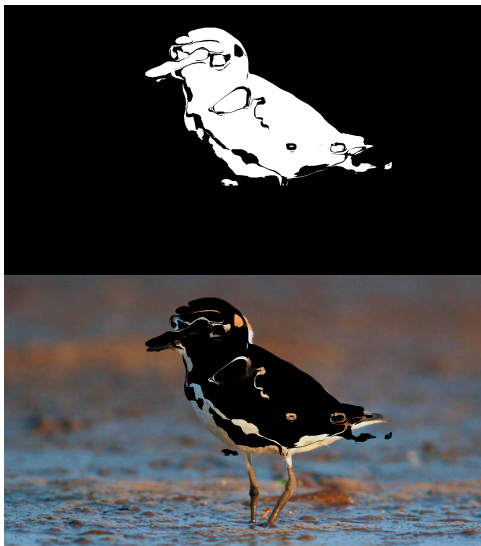


Fig. 3. Mascara y resultado.

#### IV. ANÁLISIS Y CONCLUSIONES

En el dominio de la imagen se implementó un filtro no lineal conservador de bordes. Se obtuvieron excelentes resultados con una máscara móvil de 11x11 píxeles, sin embargo el tiempo de procesamiento es excesivo (alrededor de 7 minutos por iteración). Otra dificultad que tiene este método consiste en la variedad de colores del objeto a segmentar. Mientras mayor sea la variedad de los colores que tiene el objeto y éstos se asemejen a los colores presentes en el entorno, mayor cantidad de sub-máscaras de segmentación se requieren. Cada sub-máscara a su vez se define por ensayo y error para que funcione correctamente.

No se logra segmentar la imagen completamente, aunque no deja de ser una opción viable de segmentación con el tiempo adecuado y número de iteraciones suficientes.

Posibles mejoras que se le puede incorporar para que sea más competente en cuanto a tiempo y eficiencia podrían ser mezcla con otros métodos de detección de bordes y lógica combinatorial para eliminar los "huecos de segmentación" que se pueden apreciar en la figura 3.

#### REFERENCES

- [1] W. Burger and M. J. Burge, *Edge-Preserving Smoothing Filters*. London: Springer London, 2016, pp. 413–451. [Online]. Available: [https://doi.org/10.1007/978-1-4471-6684-9\\_17](https://doi.org/10.1007/978-1-4471-6684-9_17)



**Daniel Acuña** Estudiante de Ingeniería UC, Mayor en Ingeniería en Robótica, Minor en Ingeniería en Sistemas de Automatización. Pontificia Universidad Católica de Chile.