

Práctica 2 – Visión Artificial 22/23

Introducción:

Dada la naturaleza de las imágenes con 3 canales de color, mi primera aproximación al problema fue trabajar sólo con los valores de iluminación, puesto que el tono y la saturación de las imágenes no me aporta ninguna información útil.

Para esto extraje el tercer canal de las imágenes una vez convertidas al formato HSV, mas en este punto se me plantean dos alternativas de conversión nativas de OpenCV, RGB2HSV o BGR2HSV.

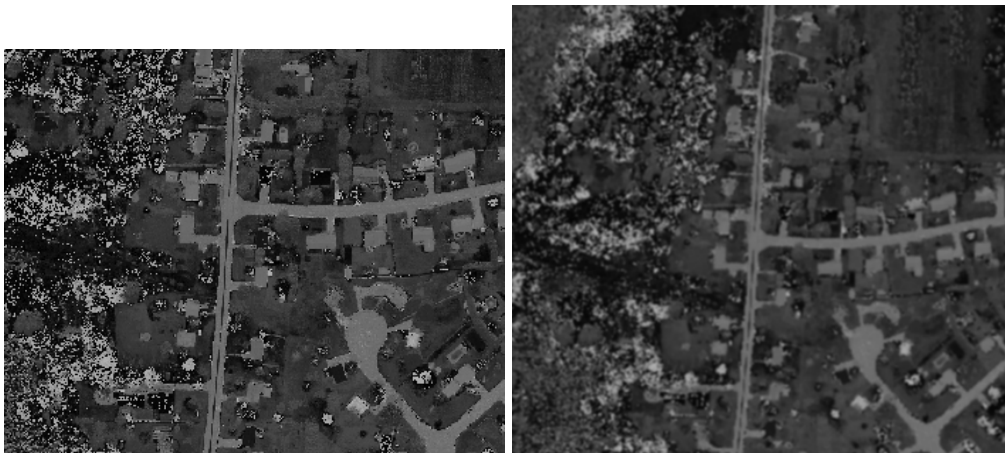


Imagen transformada de BGR a HSV



Imagen transformada de RGB a HSV

El siguiente paso fue el de aplicar un suavizado gaussiano para eliminar el ruido impulsional generado en la transformación de formato de la imagen.



Detalles de una imagen antes y después de ser suavizada.

Una idea en este punto del proceso fue aunar imágenes en escala de gris en formato HSV con otras en formato BGR, pero la descarté porque producía irregularidades que dificultaban la umbralización, las cuales no fui capaz de solventar.

Otra idea en este punto fue trabajar con imágenes después de haber ecualizado su histograma, pero una vez más, a efectos de facilitar la umbralización, fue descartada.

El primer paso después de suavizar la imagen es umbralizarla.

En este caso se emplea un umbral específico para cada imagen, el cual se calcula sumando del máximo valor de la imagen con el mínimo y dividiendo todo esto entre una constante de 2.5.

La razón de esta umbralización se debe a que produce un umbral lo suficientemente holgado para no perder demasiada partes de las carreteras.

Inicialmente tenía la idea de explorar una umbralización adaptativa, usando una ventana de partes grandes de la imagen, pero finalmente lo descarté por necesidad de probar con muchos parámetros de tamaño de ventana y C (parámetro de la función de umbralizado adaptativo de OpenCV) hasta llegar a uno óptimo.

Lo siguiente después de esto es eliminar las formas demasiado pequeñas en la imagen como para ser carreteras. Esto se hace aplicando una apertura a cada imagen para luego erosionarla.

Para eliminar las formas demasiado grandes se sustrae de la imagen la apertura con un kernel relativamente alto (15x15).

Para terminar se aplican varias transformadas Hit-or-Miss a cada imagen en un proceso iterativo: se suman el resultado de la transformada con un kernel horizontal, otro vertical y otros dos diagonales, esto con varios tamaños y grosores para cada kernel. Esto nos permite quedarnos con formas más rectas y finas.

Debido a que el rango de kernels de la transformada Hit-or-Miss no es todo lo bueno que podría muchas imágenes con carreteras más curvas o imágenes con cierta rotación producen resultados no satisfactorios.

La estimación de la longitud de las carreteras presentes en la imagen se realiza contando el número de píxeles blancos. Dado que los resultados de este algoritmo tienen una fuerte presencia de ruido, esta métrica no es del todo fiable.