

Práctica 2

Filtrado de imágenes en el dominio espacial

Curso 2019 – 2020

El objetivo de esta práctica es familiarizar al alumno con las herramientas básicas de filtrado espacial utilizando MATLAB.

Utilice la ayuda de MATLAB (`help+` comando) para conocer el funcionamiento de los comandos utilizados en este guión. Tenga en cuenta que a lo largo de la práctica puede utilizar las instrucciones `clear all` y `close all` para evitar posibles interferencias con otras variables o ventanas.

En esta práctica trabajaremos con dos imágenes en escala de grises (*gray scale*):

- Una imagen sintética (artificial) que creará el propio alumno.
- La imagen 'coins.png'.

I. Imágenes contaminadas con ruido

Para observar la efectividad del filtrado espacial conviene trabajar con imágenes reales o con imágenes sintéticas contaminadas con alguna fuente de ruido. MATLAB permite degradar la calidad de una imagen con ruido de distinta naturaleza mediante la función `imnoise`. En esta práctica trabajaremos con tres fuentes de ruido: 'gaussian', 'salt & pepper' y 'speckle'.

Considere las fuentes de ruido 'gaussian' y 'speckle' (también denominados ruido gaussiano y ruido granular, respectivamente) y contamine con ellas una imagen sintética de 8 bits, intensidad uniforme y luminancia 128. Considere que la imagen es cuadrada y de tamaño 256x256 píxeles. Indique y justifique la secuencia de instrucciones utilizada, tanto para generar la imagen como para añadir ruido. Tenga en cuenta que la matriz que representa la imagen debe ser de tipo `uint8`. Considere que las fuentes de ruido tienen media nula y su potencia media es 0.02 en los dos casos.

Justifique los resultados obtenidos en las imágenes a partir de la comparación del histograma de todas las imágenes. En la **representación del histograma**, considere en todos los casos selección automática del rango en el eje vertical.

¿Qué sucede con el histograma si la fuente de ruido considerada es de tipo 'salt & pepper'? Utilice también el valor 0.02 como la probabilidad de que un píxel sea ruidoso cuando se considera el ruido impulsivo. Indique la secuencia de instrucciones utilizada. Represente la imagen y el histograma correspondiente, justificando las diferencias con las imágenes e histogramas anteriores.

II. Filtros espaciales suavizadores

Filtrado lineal

Si un operador local efectúa una transformación lineal, su comportamiento puede definirse completamente mediante la respuesta $h[n,m]$ al impulso. Si $h[n,m]$ es rectangular y simétrica respecto de su origen, su aplicación sobre la imagen (es decir, la operación de convolución) puede efectuarse aplicando la máscara $w[n,m] = h[-n,-m]$ sobre cada píxel de la imagen original.

MATLAB ofrece la función `imfilter` para llevar a cabo la operación de filtrado lineal, para lo cual toma como parámetros la imagen original y la máscara de filtrado. La imagen resultante es del mismo tipo que la imagen original (`double`, `uint8`, `uint16`, etc.).

Teniendo en cuenta lo anterior, defina una máscara de suavizado de tamaño 5x5 que garantice que no se modifica el rango dinámico de la imagen. Indique y justifique la secuencia de instrucciones utilizada.

Filtre la imagen sintética contaminada con ruido de naturaleza gaussiana con la máscara generada en el párrafo anterior. Utilice para ello la función `imfilter` con dos argumentos de entrada (imagen y máscara): en este caso, `imfilter` genera una imagen del mismo tamaño que la imagen a filtrar y aplica la opción de tratamiento de bordes conocida como 'zero padding'. Después, repita el filtrado considerando la opción de filtrado 'symmetric' ('mirror padding'). Indique la secuencia de instrucciones utilizada en cada caso, representando las imágenes filtradas y sus histogramas. Comente y justifique las similitudes y diferencias (tanto en las imágenes como en los histogramas asociados) teniendo en cuenta las opciones de filtrado consideradas, en especial las asociadas a 'Boundary options'.

Repita el procedimiento del párrafo anterior con la opción de filtrado 'symmetric' y las fuentes de ruido de tipo 'salt & pepper' y 'speckle'.

Justifique razonadamente qué efecto tendrá aumentar el tamaño de la máscara. Apoye su justificación en un ejemplo práctico considerando una máscara de tamaño 35x35 y la imagen sintética contaminada con ruido gaussiano. Obtenga el histograma de la imagen filtrada. Comente las diferencias con la imagen filtrada que usa una máscara de tamaño 5x5, así como con el histograma correspondiente.

Filtrado no lineal

Uno de los filtros no lineales más utilizados es el filtro de mediana, que asigna a cada píxel de la imagen procesada la mediana de los valores de los píxeles situados en un entorno local. MATLAB permite aplicar este tipo de filtrado con el comando `medfilt2`.

Utilice una máscara de tamaño 5x5 y aplique un filtro de mediana sobre todas las imágenes sintéticas contaminadas con ruido. Considere como opción de filtrado 'symmetric'. Indique la secuencia de instrucciones utilizada. Comente y justifique los resultados, considerando tanto las imágenes filtradas como sus correspondientes histogramas. Compare los resultados con los obtenidos al aplicar el filtro lineal. Extraiga conclusiones.

III. Filtros espaciales de realce de contornos

En esta última etapa utilizaremos la imagen de monedas 'coins.png'. Lea la imagen y almacénela en la variable `I`.

El comando `fspecial` permite crear máscaras de filtrado espacial de determinados tipos. Haga uso de la ayuda de este comando para crear una máscara `H` de tamaño `3x3` que implemente el filtro de Prewitt. Se pide:

- filtrar la imagen 'coins.png' con la máscara `H`;
- enfatizar los contornos de `I` ortogonales a los anteriores modificando la máscara de filtrado (aunque ésta debe seguir siendo de tipo Prewitt);
- utilizar las dos imágenes filtradas para construir una imagen que aproxime el módulo del gradiente. Almacene la imagen obtenida en la variable `I_grad_Prewitt`.

Incluya la secuencia de instrucciones utilizada para obtener la imagen `I_grad_Prewitt`, así como la propia imagen.

Umbralice (comando `im2bw`) la imagen `I_grad_Prewitt` para obtener una imagen binaria donde los píxeles de primer plano correspondan a los contornos de las monedas. ¿Considera adecuado el resultado? Justifique su respuesta.

Utilice ahora una máscara espacial diferente, también de tamaño `3x3`, pero asociada a un filtro isotrópico de realce de contornos como el estudiado en las clases teóricas. Aplique el filtro a la imagen `I` y umbralice el resultado para obtener una imagen binaria donde los píxeles de primer plano correspondan a los contornos de las monedas. ¿Considera adecuado el resultado? Justifique su respuesta.

¿Qué filtro de realce de contornos, de los dos anteriores (Prewitt/isotrópico), considera más adecuado para obtener los contornos de las monedas? En lo que sigue utilizaremos ese filtro, pero aplicando una etapa de preprocesado a la imagen `I` para tener una versión suavizada de la misma. Recuerde cuáles son los dos tipos de filtro de suavizado utilizados en esta práctica y justifique razonadamente cuál de los dos considera que es más adecuado utilizar para obtener posteriormente los contornos de las monedas. Una vez elegido el filtro, utilícelo con una máscara de tamaño `11x11` para suavizar la imagen `I`, y umbralice la imagen resultante de modo que todos los píxeles asociados a las monedas sean píxeles de primer plano, y todos los píxeles de fondo correspondan a la superficie sobre la que se encuentran las monedas. Proporcione una imagen binaria de tipo `uint8` donde el valor `255` corresponda al primer plano, guardando la imagen resultante en la variable `I_BW`. Indique la secuencia de instrucciones utilizada y la imagen binaria resultante.

Aplique el filtro de realce elegido previamente a la imagen `I_BW` y almacene el resultado en la variable `I_BW_realce`. ¿Qué obtiene?

IV. Composición de imágenes

En la última parte de esta práctica haremos una composición de las imágenes `I` e `I_BW_realce` para obtener una imagen RGB similar a la que se muestra en la Figura 1. Proponga el procedimiento a seguir e impleméntelo en MATLAB. Anote la secuencia de instrucciones utilizada, teniendo en

cuenta que, en la medida de lo posible, es importante que las imágenes generadas sean de tipo uint8. Puede resultar conveniente hacer uso del comando `imadd`, que permite realizar la suma de dos imágenes del mismo tipo.

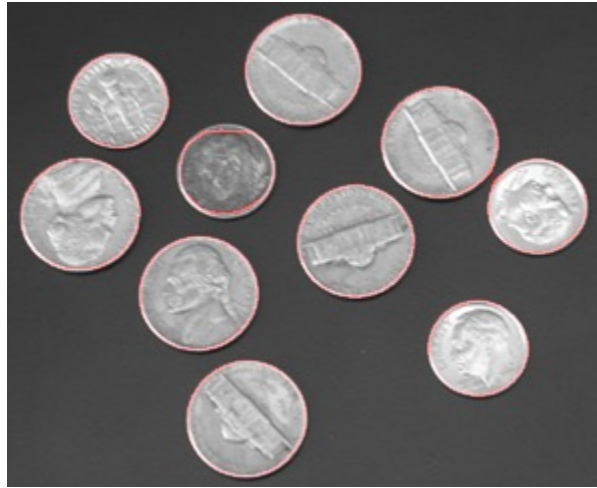


Figura 1. Imagen RGB con monedas enmarcadas por curvas de trazo continuo y color rojo.