
Laboratorio 2: Fundamentos de Procesamiento Digital de Imágenes

Profesor Henry Arguello Fuentes, henarfu@uis.edu.co

21 de octubre de 2016

El informe del laboratorio debe entregarse el día 28 de octubre, en éste deben responder las preguntas que se presentan en la guía, además anexar los archivos que generen, por ejemplo .mat, .m, .fig, etc. Enviar una carpeta comprimida con los archivos al correo electrónico: jonathan.monsalve@saber.uis.edu.co, con el asunto **Laboratorio 2 imágenes** y el nombre de la carpeta comprimida **Lab2_NombreApellido**, donde Nombre es su primer nombre y Apellido es su primer apellido.

1. Objetivo

Comprender los conceptos de Cuantificación, y submuestreo. Y el ruido asociado a estos conceptos.

2. Cuantificación

Cuantificación se puede definir como un mapeo de una señal de un dominio en el espacio \mathbb{R} a un espacio \mathbb{N} . Lo que provoca que la diferencia entre 2 niveles cuantificados sea un paso de tamaño definido (*quant*). La relación de el *quant* y el número de bits que se necesitan para la cuantificación es:

$$q = \frac{A}{2^B - 1} \quad (1)$$

donde *q* hace referencia al *quant* y B al número de bits.

2.1. Problemas

- ¿Qué hace la función de MATLAB *quant()*? Como podemos usarla para uniformemente cuantificar una imagen con N niveles?
- Cree una función para cuantificar una imagen I, de tal manera que el rango dinámico de la imagen sea cuantificado en N niveles uniformes. Donde N es un parámetro de entrada de su función (al igual que la imagen I).
- use una imagen en escala de grises. Muestre dicha imagen y las imágenes cuantificadas usando desde 1 hasta 8 bits.
- ¿Cuántos bits se necesitaron para que note la diferencia?
- ¿Qué es el ruido de cuantificación?, ¿Cuál es el máximo valor posible de ruido de cuantificación para los ejemplos anteriores?

3. Muestreo

El muestreo es un proceso mediante el cual se crea una señal discreta a partir de una continua, de tal forma que los valores (muestras) son tomados en ciertas posiciones. si las muestras se toman con el mismo espacio, el muestreo se llama uniforme. Dado que los computadores no pueden almacenar señales continuas. trabajaremos con señales

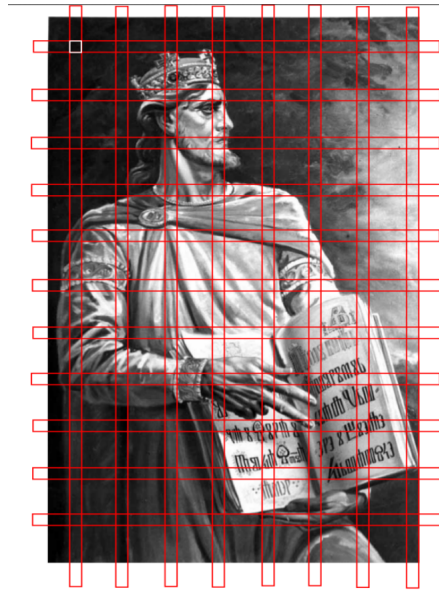


Figura 1: El muestreo toma unicamente los valores de los pixeles que se interceptan en las lineas verticales y horizontales.

densamente muestreadas.

3.1. Ejemplo

```
>> [img,map]=imread('coins.png');  
>> coins =img(1:10:end,1:10:end,:);
```

3.2. Problemas

- submuestree una imagen con factor 1 a 10. Muestre y comente los resultados.
- Seleccione una imagen, de esta seleccione un objeto que desea censurar y usando el comando *imresize* a este pequeño objeto reduzca y aumente su tamaño varias veces y luego ponga el objeto en su lugar. esto generará que dicho objeto se pixele. el resultado será algo como esto.

