

Profesor: Antonio J. Caamaño Fernández. antonio.caamano@urjc.es
Profesor de Prácticas: Eduardo del Arco Fernández. eduardo.delarco@urjc.es

1. Descripción y objetivos

Se trata de analizar las prestaciones de distintos cuantificadores y codificadores. La fuente de información es una señal de audio digital PCM. En primer lugar, sometemos a la señal a un filtro paso bajo digital FIR cuyo funcionamiento está fuera del temario de la asignatura ¹. Tras limitar la señal en banda, se ensayarán tres cuantificadores y codificadores y dos tipos de codificadores entrópicos. Los cuantificadores son:

1. *Cuantificador uniforme*: Es un cuantificador escalar. Los niveles de cuantificación adyacentes están a la misma distancia. En la presente práctica podemos observar los efectos al variar el número de niveles de cuantificación y el margen dinámico del cuantificador.
2. *Cuantificador uniforme con codificación DPCM*: Es una codificación mediante un predictor lineal. En esta práctica, realizaremos un predictor de primer orden. La salida del codificador es la diferencia ponderada de dos muestras consecutivas. Tal como se verá, este sistema consigue ganancias de codificación elevadas, entre 2 y 4, respecto al cuantificador uniforme.
3. *Cuantificador vectorial no uniforme*: Existen muchos cuantificadores de este tipo. En esta práctica se ha realizado un clasificador mediante el algoritmo *kmedias*. No entraremos en los detalles del algoritmo, pero sí es interesante conocer cualitativamente cómo funciona un cuantificador vectorial. Se trata cuantificar y codificar bloques de muestras, no muestras individuales. De ahí la denominación *vectorial*. Es un cuantificador no uniforme porque la distancia entre niveles de cuantificación no es regular.

Respecto a los codificadores, se ensayarán dos:

1. *Codificación Huffman*. Este codificador es de longitud variable y requiere un conocimiento *a priori* de la ocurrencia de los símbolos. De esta forma, se asignan menos bits a los símbolos más probables. El codificador y decodificador deben compartir un diccionario de códigos. Consigue buenos resultados cuanto menos equiprobables sean los símbolos.
2. *Codificación LZW*. Este codificador es de longitud fija y es utilizado universalmente. Las palabras código se van armando a medida que los símbolos se suceden, aprovechando los patrones que entre éstos pudieren existir. Por lo tanto, no es necesario un conocimiento *a priori* de la secuencia. El diccionario de palabras código se elabora sobre la marcha y el decodificador no lo necesita para empezar a decodificar.

La presente práctica queda definida según el esquema 1.

¹No obstante, si desea conocer con mayor profundidad cómo funciona un filtro de estas características el profesor estará encantado de explicárselo

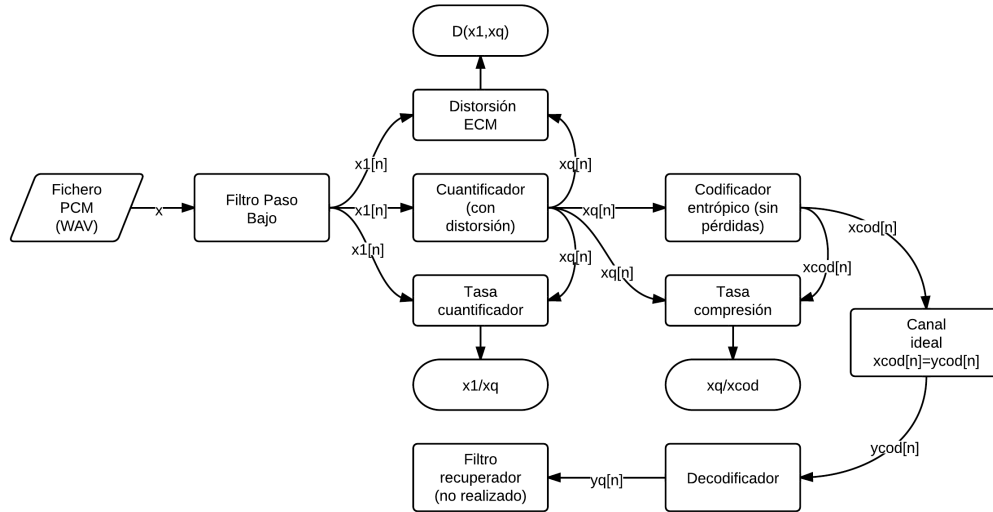


Figura 1: Esquema general de la práctica

2. Desarrollo de la práctica

2.1. Nota previa

El *script* está dividido en sub apartados independientes que pueden ejecutarse por separado utilizando modo *cell*. El fragmento de código activo aparece levemente sombreado en color amarillo y se ejecuta pulsando el botón *evaluate current cell*. (Véase figura 2). Coloque el cursor al principio del documento, aparecerá seleccionado y listo para ejecutarse.

2.2. Parte común: Procesado previo de señal

Edite el archivo *uniforme.m* y vaya ejecutando cada bloque de acuerdo con la siguiente enumeración:

- 1) *Apertura de archivo y procesado*: Se lee el archivo de audio . El archivo original tiene 18 segundos de duración, dos canales, una frecuencia de muestreo de 44.1 kHz y 16 bits por muestra. Para hacerlo más manejable, tomaremos un fragmento de unos 5 segundos, eliminaremos uno de los canales y someteremos la señal a remuestreo mediante un filtro FIR y diezmado de orden 4. Este procedimiento está totalmente automatizado en el *script*. El resultado es una secuencia a 11025 Hz, con 16 bits por muestra.
- 2) *Espectrograma*: Se trata de una representación tiempo-frecuencia que no forma parte de la asignatura. No obstante, resulta interesante ver y escuchar qué tipo de señal vamos a cuantificar y codificar.

2.3. Cuantificación uniforme

- 3) *Parámetros del cuantificador*: Es posible cambiar dos parámetros del cuantificador, el número de niveles y el margen dinámico. El *script* genera una representación de la forma de onda original y la cuantificada. Por el momento, deje los parámetros tal como están, observe atentamente la gráfica y pase al siguiente punto.

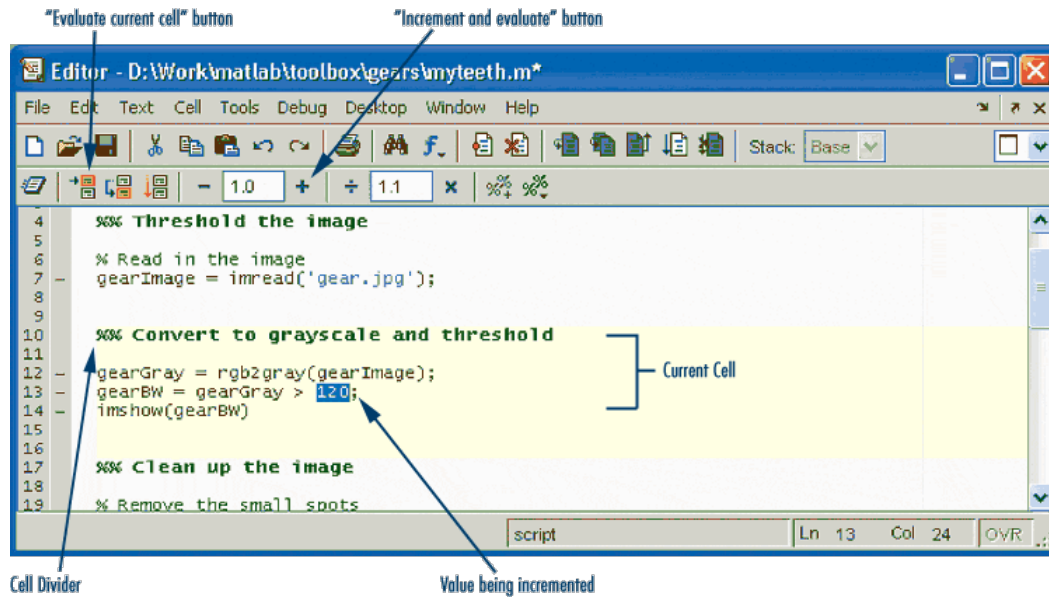


Figura 2: Ejemplo de código en modo *cell* con explicación del significado de las funciones más relevantes. En esta práctica se utilizará el modo *cell* para ejecutar bloques separados, pero no para iterar los mismos.

- 4) *Evaluación de las prestaciones del cuantificador*: Ejecute el *cell*. Escuche atentamente la grabación y Observe las gráficas. Haga uso del zoom para ver los efectos de la cuantificación. Observe que el error cuadrático se mantiene aproximadamente constante salvo en un punto, donde aumenta varios órdenes de magnitud. Responda a las siguientes preguntas:

- a) ¿Por qué existe la distorsión en la señal cuantificada y cómo se está midiendo?
- b) ¿Por qué se produce un aumento repentino del error en un punto concreto?
- c) La falta de claridad del sonido, ¿a qué se debe?

- d) Cambie los parámetros por otros que considere oportunos. Observe y escuche el resultado. Elabore un cuadro con el error cuadrático medio para las siguientes parámetros $M_{din} = \{0, 5, 0, 7, 1, 1, 3, 1, 7\}$ y $N_{niveles} = \{4, 8, 16, 32, 64\}$. ¿Puede proporcionar un valor óptimo para los parámetros? Escriba una pequeña reflexión al respecto.
- f) Utilice MATLAB y para calcular la ganancia del cuantificador para los casos anteriores, suponiendo que todos los niveles de cuantificación se codifican con el mismo número de bits. CONSEJO: Averigüe cómo se calcula la ganancia pedida y realice ambos apartados juntos.

- 5) *Codificador y decodificador Huffman*: Ejecute el *cell* y compruebe que las secuencias x_q e y_{cod} son iguales. Observe las secuencias generadas por el codificador y el diccionario. Responda:
- a) ¿Qué relación existe entre la gráfica *Ocurrencia de los niveles de cuantificación* y el diccionario que el codificador ha generado?
 - b) Ensaye distintos niveles de cuantificación y explique por qué Huffman consigue mejores prestaciones con pocos niveles o con muchos niveles.
 - b) Utilice MATLAB para calcular la ganancia de codificación mediante Huffman, para 64 niveles y margen dinámico 1,25.
 - c) Utilice los resultados anteriores para calcular la ganancia total x_1/x_{cod} .

- 6) *Codificador y decodificador LZW*: A continuación se realiza una codificación LZW como la que se ha visto en clase de teoría. Observe que la secuencia LZW se puede decodificar sin necesidad de utilizar un diccionario. El libro de códigos es generado en el receptor al mismo tiempo que se realiza la decodificación. Responda a las siguientes preguntas:
- a) Calcule la ganancia de compresión LZW para 64 niveles y 512 niveles, ambas con un margen dinámico de 1,25.
 - b) Con 512 niveles, compare la velocidad de Huffman con LZW. ¿Cuál es más rápido? ¿Por qué?
 - c) ¿Con cuál se consiguen mejores prestaciones? ¿Por qué?
 - d) Compare la ganancia total de ambos códigos, para estos últimos valores. Explique el resultado.

2.4. Cuantificación uniforme y codificación DPCM

En el apartado anterior se ha mostrado un cuantificador uniforme sin memoria, ya que la cuantificación está basada en la muestra actual y nada más. En DPCM, se realiza una predicción de la siguiente muestra a partir de muestras del pasado. Esta predicción es utilizada por el cuantificador para afinar el intervalo de cuantificación. Así, la región de incertidumbre se reduce gracias a la redundancia natural de la señal. Esta redundancia se identifica y se reduce, restándola de la predicción de la siguiente muestra. Es lo que se denomina *error de predicción*. La ecuación del predictor es:

$$\hat{x}[n] = ax[n-1] \quad (1)$$

Donde a es el parámetro utilizado para minimizar el *error de predicción*. El cálculo de a es intuitivo: Decidir a para minimizar una función de coste. Como siempre, la función de coste es el error cuadrático medio. El error, o distorsión, se define como

$$d[n] = x[n] - ax[n-1] \quad (2)$$

El error cuadrático medio entonces es:

$$E[d^2[n]] = E[x^2[n] + a^2x^2[n-1] - 2ax[n]x[n-1]] \quad (3)$$

Reescribimos la anterior expresión:

$$R_d[0] = R_x[0][1 + a^2 - 2a \frac{R_x[1]}{R_x[0]}] \quad (4)$$

Donde $R_d[0]$ es la autocorrelación en 0 del error de predicción y $R_x[n]$ es la autocorrelación de la señal de entrada. Derivamos respecto a a e igualamos a 0 para encontrar el óptimo:

$$a_{opt} = \frac{R_x[1]}{R_x[0]} \quad (5)$$

Así, el parámetro óptimo para el predictor lineal de primer orden depende de la autocorrelación normalizada en $n-1$. Es necesario calcular la autocorrelación de la señal de entrada. Volviendo a MATLAB, edite el archivo *dpcm.m* y vaya ejecutando cada bloque de acuerdo con la siguiente enumeración. Los dos primeros apartados son idénticos a la cuantificación uniforme. Ejecutelos y focalice el tercer *cell*.

- 3) *Cuantificación y DPCM*: Es posible variar los parámetros del cuantificador y también el parámetro a del predictor. El parámetro óptimo es el que minimiza el error cuadrático medio. Para ello observe como calculamos la autocorrelación de la señal de entrada. Responda a las siguientes preguntas:

- a) Haga un esbozo de la función de autocorrelación.
- b) ¿Cual es el rango de a ?

- 4) *Evaluación de las prestaciones del cuantificador*: Ejecute el *cell*. Escuche atentamente la grabación y Observe las gráficas. Haga uso del zoom para ver los efectos de la cuantificación. Responda a las siguientes preguntas:
- a) Deje los parámetros de cuantificación constantes y ensaye varios valores de a dentro del rango permitido. Observe el error cuadrático medio.
 - b) Tanto si ha utilizado la función de autocorrelación o ha encontrado el parámetro óptimo probando diferentes valores de a , repita los apartados **e** y **f** y compare las cifras obtenidas con las de cuantificación uniforme.
 - c) ¿Por que ha desaparecido hasta un nivel despreciable el error por sobrecarga?
- 5) *Codificador y decodificador Huffman*: Ejecute el *cell* y compruebe que las secuencias antes de codificar y después de decodificar son iguales. Observe las secuencias generadas por el codificador y el diccionario. Responda:
- a) Utilice MATLAB para calcular la ganancia de codificación mediante Huffman, para 64 niveles y margen dinámico 0,5. Comparelo con los resultados del apartado de cuantificación.
 - c) Utilice los resultados anteriores para calcular la ganancia total x_1/x_{cod} .

- 6) *Codificador y decodificador LZW*: A continuación se realiza una codificación LZW como la que se ha visto en clase de teoría. Observe que la secuencia LZW se puede decodificar sin necesidad de utilizar un diccionario. El libro de códigos es generado en el receptor al mismo tiempo que se realiza la decodificación. Responda a las siguientes preguntas:
- a) Calcule la ganancia de compresión LZW para 64 niveles y 512 niveles, ambas con un margen dinámico de 0,5.
 - b) Con 512 niveles, compare la velocidad de Huffman con LZW. ¿Por qué es más rápido que con la cuantificación uniforme?
 - c) ¿Con cuál se consiguen mejores prestaciones? ¿Por qué?
 - d) Compare la ganancia total de ambos códigos, para estos últimos valores. Explique el resultado. Haga una comparación con las cifras que obtuvo con la cuantificación uniforme y discuta el resultado.

2.5. Cuantificación vectorial no lineal

Explicacion en pizarra.