

Práctica 1 - TDS

Señales de voz: ver, oír, grabar y procesar

Práctica 1

OBJETIVOS

1. Señales de sonido con extension “*.wav”. Cargar y reproducir sonido en Matlab
2. Representación en tiempo discreto y continuo de señales conociendo la frecuencia de muestreo
3. Procesado digital de señales por tramas (*frames*). Cálculo de la energía de una señal por tramas
4. Para casa: Grabar sonido usando Matlab ¿Cuál es el *pitch* de tu voz?

Importante:

- Se proporciona el fichero mi_practica1.mlx donde escribir el código de Matlab utilizado para cada ejercicio.
- **Este fichero se debe entregar antes de la siguiente sesión junto con su impresión digital en pdf** (se puede hacer desde Matlab). Se valorará que el fichero tenga comentarios explicativos.
- En las siguientes transparencias, el texto de color **marrón** indica que son variables o instrucciones para usar en Matlab

Ejercicio 1.1

- Carga el fichero sound1.wav en Matlab. Observa las variables que crea en el Workspace

```
>> [y1,fs] = audioread('sound1.wav');
```

- Busca como usar `sound` y `soundsc` ¿Qué oyes? Etiqueta cada sonido
- Representa la señal en tiempo discreto creando el vector `n1`:

```
>> Ly1 = length(y1); % Ly1 es la longitud de y1
```

```
>> n1=(0:Ly1-1); % n1 tiene el mismo número de elementos que y1
```

```
>> plot(n1,y1), xlabel('n (discreto)')
```

- Sabiendo que `y1` es una señal muestreada a frecuencia de muestreo (*sampling rate*) `fs`, escribe una función de Matlab llamada `dibuja.m` tal que al ejecutar `dibuja(y1,fs)`, se represente en una figura nueva la señal `y1` con el eje temporal en segundos
- Para cada fichero wav proporcionado, realiza los mismos pasos que para 'sound1.wav':
 - Escucha el sonido y etiquétalo
 - Dibuja la señal en tiempo discreto y en tiempo continuo (usando `dibuja.m`) en dos figuras distintas.

Ejercicio 1.2

- En este ejercicio y el siguiente usaremos solo los sonidos almacenados en **y1**, **y2** e **y3**. Crea los vectores **yframe1**, **yframe2** e **yframe3** de forma que cada uno de ellos contenga un tramo de 100ms de la señal correspondiente en donde aparezca sonido. Dibújalos en figuras independientes con el eje temporal en segundos (usa **dibuja.m**)
- ¿Cuáles tienen estructura periódica? A partir de lo que ves, estima cuál es el periodo fundamental de las que son periódicas, **T**, y calcula su frecuencia fundamental, **f0**, como el inverso de **T**.
- Para las señales completas **y1**, **y2** e **y3**: ¿cuál es la potencia media, **P**, de cada una de ellas?

$$P = \frac{1}{L_y} \sum_{n=0}^{L_y-1} y^2(n)$$

- ¿Y cuál es la energía, **E**, de cada una de ellas? $E = \sum_{n=0}^{L_y-1} y^2(n)$

Procesado digital por tramas (*frames*)

- La energía de cada señal depende del valor de las muestras pero también de si la señal es más larga o más corta. En la vida real las señales son muy largas y diversas.
- Las señales de sonido se suelen procesar por tramas de muestras (*frames*), de tal manera que el tamaño de la trama es siempre el mismo.
- Su tamaño se puede expresar en número de muestras (*samples*) o en segundos. Por ejemplo, si queremos un tamaño de trama de 20ms, podemos calcular el número de muestras que tendrá sabiendo la frecuencia de muestro *fs*:

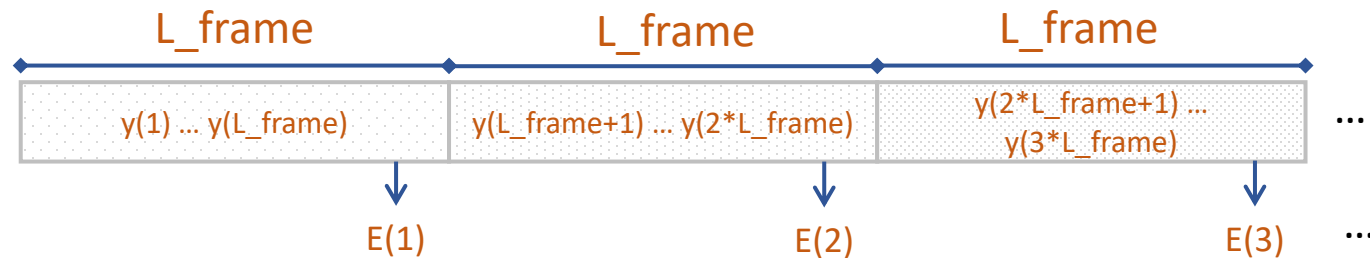
```
>> L_frame=round(0.02*fs); % usamos round para que L_frame sea entero
```

- Si trabajamos a *fs*=16 kHz, la trama tiene un tamaño *L_frame*=320 muestras. Por otro lado, si nos dicen que la trama tiene 512 muestras, podemos saber su duración, haciendo la operación inversa:

```
>> L_duracion=512/fs; % 32ms con fs=16kHz
```

Ejercicio 1.3

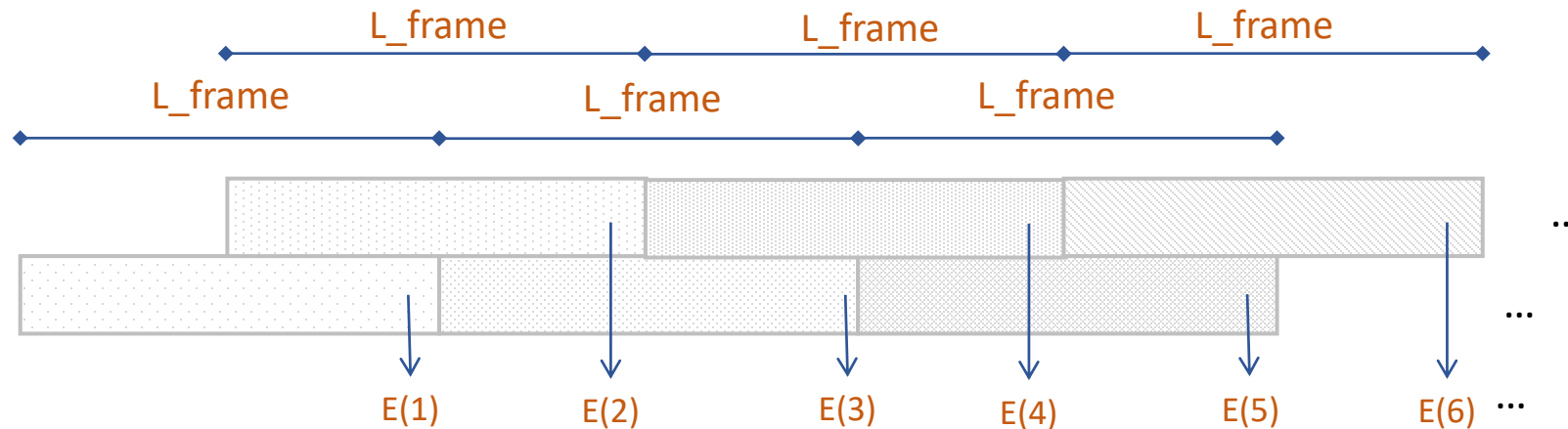
- Calcula la energía de las señales **y1**, **y2** e **y3** por tramas de 100ms y almacena el resultado en un vector **E**, distinto para cada señal:



- Piensa como puedes hacerlo **NO USANDO BUCLES** (FOR, WHILE, REPEAT,...)
- La longitud de cada vector **E** es distinta porque los sonidos también tienen distinta duración. Se pueden dibujar vectores de distinta longitud en una misma figura usando **hold**. Busca en la documentación de Matlab como se usa **hold** y dibuja las tres energías en una misma figura con un eje de tiempos discreto. Comenta la diferencia entre ellas.
- Dibuja ahora las tres energías en un eje de tiempos en segundos. Ten en cuenta que obtienes una ‘muestra’ de la energía cada 100ms, por lo que tienes que crear un eje de tiempos en segundos distinto del usado para las voces.

Ejercicio 1.4

- Generalmente las muestras seleccionadas en la trama m -ésima no son todas nuevas, sino que se seleccionan algunas muestras de la trama anterior ($m-1$) y el resto son muestras nuevas. En la figura se han seleccionado la mitad ($L_frame/2$) muestras de la trama anterior y ($L_frame/2$) muestras nuevas. Se dice entonces que el solape (**overlap**) es del 50%.

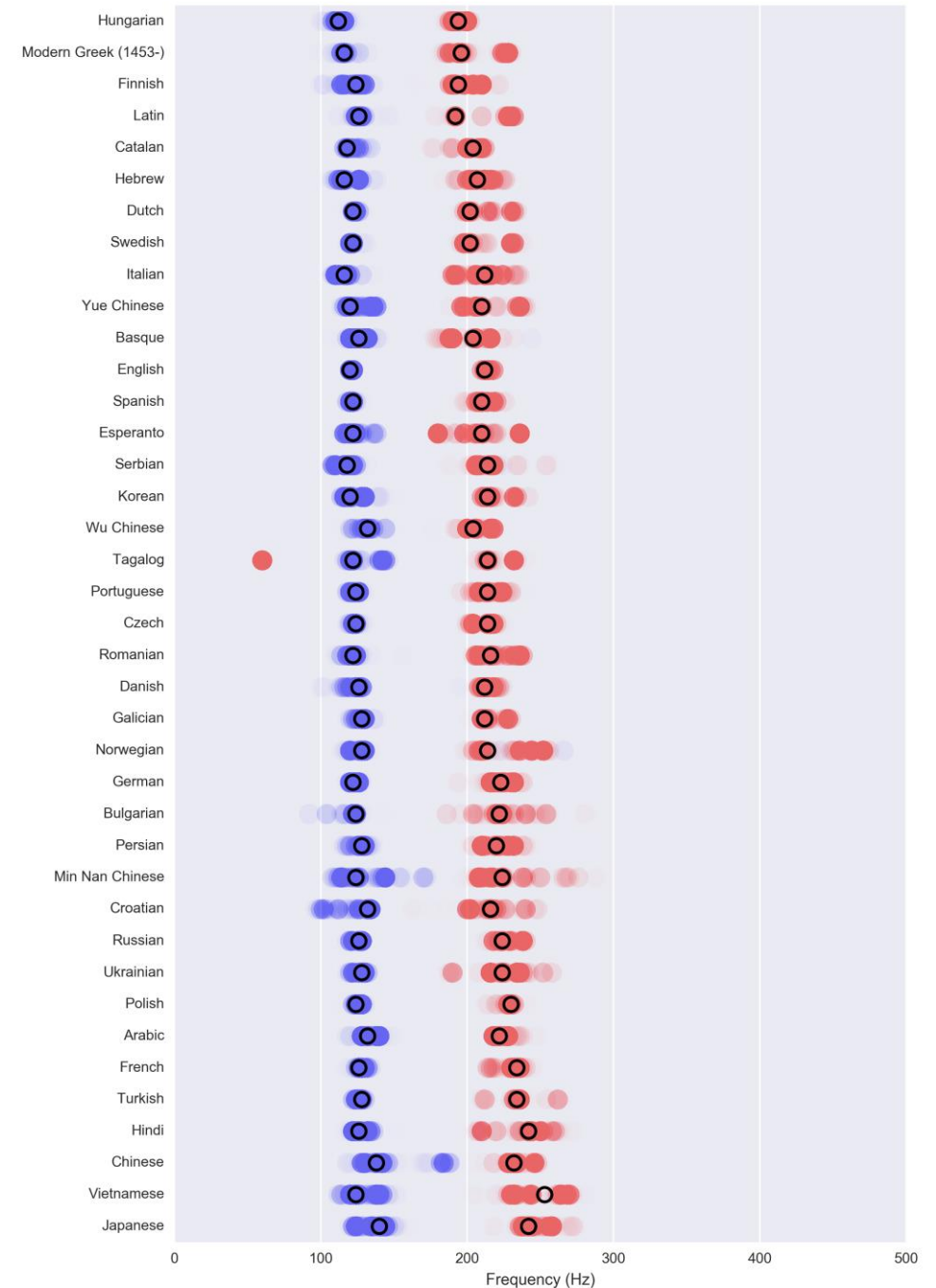


- Calcula ahora la energía de las señales **y1**, **y4** e **y5** por tramas de 20ms y un solape del 50% y dibújalas en una misma figura con el eje de tiempos en segundos. Ten en cuenta que ahora obtienes una ‘muestra’ de la energía cada 10ms.
- Comenta los resultados que obtienes respecto al sonido de cada fichero. ¿Puedes identificar donde comienza el sonido solo con su energía? ¿Puedes saber si el sonido tiene un ruido de fondo?

Ejercicio 1.5

¿Cuál es tu *pitch*?

- El *pitch* de una señal de voz es la frecuencia fundamental **f0** que has calculado en el ejercicio 1.2.
- Va asociado al 'timbre' de la persona que habla, aunque cambia ligeramente según la entonación y el lenguaje.
- El *pitch* es una de las características por las que se puede identificar una persona, como la huella dactilar, el iris, etc.
- En la figura se representa el valor del pitch medio en Hz para hombres (azul) y mujeres (rojo) según el idioma.
- Graba tu voz diciendo una vocal, o los fonemas /n/, /m/ o cualquier otro fonema sonoro. Sigue los pasos del ejercicio 1.2 y podrás saber tu *pitch*.
- Si pudieras cambiar tu voz por otra ¿cuál elegirías? Estos emprendedores valencianos te la cambian: www.voicemod.net



Para la próxima práctica

- Usa el livescript **RecordAudio.mlx** proporcionado para grabar sonidos en tu propio dispositivo
- Para el próximo día de laboratorio tienes que traer grabados los números del 'cero' al 'nueve' en un único fichero *.wav
- **Cosas importantes** a tener en cuenta:
 - Procura hacer la grabación en un entorno lo más silencioso posible.
 - Cuando hables, deja una pausa de 1 segundo entre número y número.
 - Cuando visualices la señal grabada en Matlab, comprueba que el rango de valores de las muestras está comprendido en el rango (-1,1) y que no ha llegado a dichos valores en ningún momento. Si no es así, aléjate del micrófono y vuelve a grabar.
 - Por el contrario, si ves que los valores de las muestras no pasan de 0.1-0.2 en el eje de ordenadas, acércate al micrófono y vuelve a grabar hasta que alcancen al menos 0.5 (ó -0.5) en su máxima amplitud.