

## PRÁCTICA 06: Reconocimiento de palabras aisladas (algoritmo DTW)

En esta práctica vamos a aplicar el algoritmo de alineación temporal dinámico (DTW – Dynamic Time Warping):

1. Obtener los conjuntos de vectores de características correspondientes a las pronunciaciones de los números del cero al nueve. Repítalo  $n$  veces (por ejemplo,  $n = 5$ ) obteniendo así  $10 \times n$  conjuntos de vectores de características (10 números  $\times n$  repeticiones) que consideraremos los patrones de los números a reconocer. De ahora en adelante denominaremos P.
2. Implementar una versión del algoritmo DTW con sólo restricciones locales. Sean  $p$  y  $t$  los vectores de características del patrón y de test con dimensión (número de tramas)  $n$  y  $m$  respectivamente:

```
Para cada i desde i = 1 hasta i = n
    DTW[i, 0] = infinito
Para cada j desde j = 1 hasta j = m
    DTW[0, j] = infinito
DTW[0, 0] = 0

Para cada i desde i = 1 hasta i = n
    Para cada j desde j = 1 hasta j = m
        dist = distancia entre los vect. Características i y j
        DTW[i, j] = dist + minimum(DTW[i-1, j], // restricc.
                                   DTW[i, j-1], // locales
                                   DTW[i-1, j-1])

Devolver DTW[n, m]
```

3. Generar vectores de características de un conjunto de sonidos correspondientes a las pronunciaciones de las palabras cero al nueve correspondientes a los números a reconocer. Repítalo  $m$  veces, por ejemplo  $m = 10$ , de forma similar a como hizo en el paso primero. Tendremos  $10 \times m$  conjuntos de vectores de características y que denominaremos de ahora en adelante R.
4. Aplicar el algoritmo DTW implementado en el paso 2 del siguiente modo:
  - a. Suponemos que sólo disponemos de un patrón para cada número, es decir, sólo una columna de P. Comparamos todos los conjuntos de vectores de características de R con cada  $P(:,i)$  para cada  $i$  de forma individual. Se obtendrán 10 valores de error (uno por el patrón de cada número) y nos quedaremos con aquel tal que el error sea el mínimo. Generamos  $m$  matrices de confusión para mostrar los resultados.
  - b. Supongamos que ahora disponemos de  $n$  patrones para cada número, es decir, P de forma completa. Comparamos todos los conjuntos de vectores

de características de R con cada  $P(:,i)$ , tal y como se hacía en el paso anterior, y seleccionamos el patrón más votado como salida. Creamos una matriz de confusión para mostrar los resultados.

Repetir el punto 4 para los siguientes vectores de características (antes y después del proceso de normalización):

- El logaritmo de la energía + coeficientes cepstrales en la escala de Mel
- El logaritmo de la energía + coeficientes cepstrales en la escala de Mel + delta cepstrums.
- El logaritmo de la energía + coeficientes cepstrales en la escala de Mel + delta cepstrums + delta-delta cepstrums.

### **Documentación a entregar**

Se entregarán un fichero comprimido con:

- Un pequeño informe con las matrices de confusiones y un análisis de los resultados obtenidos.
- El código MATLAB generado en la práctica.
- Los ficheros con los vectores de características de los patrones (P).
- Los ficheros con los vectores de características de las señales de audio para validar el RAH (R).