

---

## TP : Dynamic Time Warping

### Objectifs

- se familiariser avec la technique Dynamic Time Warping
- mettre en oeuvre Dynamic Time Warping pour la reconnaissance de gestes



FIGURE 1 – Exemple de résultat attendu pour l'application finale. Le geste tracé par l'utilisateur est affiché en noir, le nom du geste reconnu est affiché en haut à gauche et les points du geste de référence sont affichés en orange.

## 1 Matériel à disposition

Une interface écrite en JavaFX est mise à votre disposition pour saisir des courbes en utilisant la souris. Cette interface utilise la classe `DTWGUI` pour définir la fenêtre de l'application et capturer les interactions effectuées avec la souris.

Référez-vous à cette procédure<sup>1</sup> pour utiliser JavaFX avec Eclipse.

L'interface vous permet de saisir et afficher (en noir) une courbe de référence en cliquant sur un des boutons de la souris. La courbe en orange représente le geste le plus proche qui a été reconnu dans les gestes de référence (templates).

Une classe `Matrix` est également mise à votre disposition pour faciliter la manipulation de matrices.

`Template` et `TemplateManager` permettent de charger les gestes disponibles dans `gestures.xml`. Ce fichier contient plusieurs exemples de chacun des gestes du `$! recognize`<sup>2</sup> ainsi que les gestes "papa", "maman" et "mer".

---

1. <https://www.iut-info.univ-lille.fr/~gery.casiez/R2.02/TP/TP1Layout/>

2. <http://depts.washington.edu/aimgroup/proj/dollar/index.html>

**Q 1.** Compilez le matériel à disposition et étudiez le code fourni.

Pour répondre aux questions suivantes vous aurez besoin de créer une classe DTW et de faire quelques modifications à la classe DTWGUI.

## 2 Calcul de DTW

**Q 2.** Créez une classe DTW qui calcule la matrice D vue en cours à partir de deux ensembles de points. Vous utiliserez comme métrique la distance Euclidienne entre deux points.

**Q 3.** Ajoutez lors du calcul de la matrice, le prédécesseur de  $D(i,j)$  (utilisez `couple` de `Matrix`). Cette étape est nécessaire si vous souhaitez connaître la mise en correspondance des points des deux séquences, mais inutile pour la reconnaissance de gestes.

## 3 Application à la reconnaissance de gestes

La reconnaissance de gestes consiste à réaliser un pré-traitement sur l'ensemble des points saisis par l'utilisateur puis à l'aligner avec chacun des gestes de référence en utilisant DTW pour calculer la distance entre 2 gestes. Le geste considéré comme reconnu est celui qui a la distance DTW la plus faible.

Le pré-traitement proposé ici consiste à 1) normaliser les points de manière à ce que leur boîte englobante tienne dans un carré de référence et 2) à translater le centroïde des points à l'origine.

**Q 4.** Modifiez la classe DTW pour réaliser la reconnaissance de gestes. N'oubliez pas que les gestes disponibles dans `gestures.xml` n'ont subi aucun pré-traitement. À l'issue de cette question votre application doit au moins afficher le geste reconnu dans la console, lors du relâchement du bouton de la souris. Vous pouvez en plus afficher le geste de référence reconnu en orange, comme dans l'exemple Figure 1.

**Q 5.** Testez les différents gestes fournis dans `gestures.xml` et discutez l'efficacité de la méthode suivant le type de geste.