

TP – Moteur de reconnaissance de geste

Objectifs

- Comprendre le fonctionnement d'un moteur de reconnaissance de geste simple : le reconnaître.
- Implémenter un moteur de reconnaissance de geste (à partir de la classe **Stroke.java** donnée) qui communique sur le bus logiciel IVY de manière à pouvoir l'utiliser à partir de tracé obtenu de différentes sources

Les différentes ressources nécessaires au bon fonctionnement de ce TP sont disponibles sur la page : <https://www.irit.fr/~Mathieu.Raynal/M2ihm.html>

La palette graphique envoie sur le bus IVY un message pour chaque événement provenant du dispositif de pointage. Un message de la forme **Palette:MousePressed x=arg1 y=arg2** est envoyé lorsque l'utilisateur presse un bouton du dispositif de pointage. De même, un message **Palette:MouseReleased x=arg1 y=arg2** est envoyé lorsqu'il relâche le bouton du dispositif de pointage. Les messages **Palette:MouseMoved x=arg1 y=arg2** et **Palette:MouseDragged x=arg1 y=arg2** sont respectivement envoyés lorsque le curseur est déplacé, et lorsque le curseur est déplacé tout en maintenant un bouton du dispositif enfoncé. **arg1** et **arg2** sont des valeurs entières représentant les coordonnées du pointeur du dispositif de pointage au moment où l'événement se produit.

Afin de pouvoir vérifier les informations réceptionnées sur le bus logiciel IVY, votre moteur de reconnaissance de geste affichera sur la palette graphique : dans un premier temps le tracé reçu, puis, le geste normalisé une fois toutes les étapes de normalisation effectuées.

A- Réception et affichage d'un tracé

Créez une application qui réceptionne via le bus logiciel IVY les messages de la palette graphique qui permettent de construire un tracé :

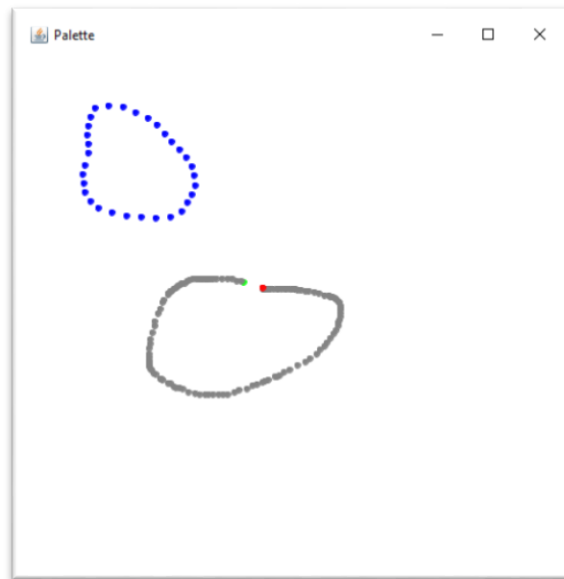
- Point de départ : MousePressed
- Tracé : MouseDragged
- Point final : MouseReleased

Votre application enverra les messages sur IVY de manière à ce que le tracé soit affiché sur la palette. Le point de départ du tracé sera affiché en vert, et le point final sera affiché en rouge. Les autres points constituant le tracé seront affichés en gris.

B- Normalisation du tracé reçu

A l'aide de la classe **Stroke.java**, appliquez au tracé les différentes étapes permettant de normaliser le geste. Une fois que le geste est normalisé, celui-ci sera affiché avec le tracé

original sur la palette graphique. Sur la figure ci-dessous, le tracé original et celui normalisé sont affichés respectivement en gris et en bleu.



C- Enregistrement et reconnaissance d'un nouveau geste

Votre application doit proposer deux modes :

- un mode « Apprentissage » où le geste effectué sera associé à une commande ;
- un mode « Reconnaissance » qui permettra de déterminer à quelle commande correspond le geste qui vient d'être effectué.

Pour le mode « Apprentissage », l'utilisateur devra pouvoir associer un nom de commande au dernier tracé reçu, avant que celui-ci ne soit associé au dictionnaire.

Pour le mode « Reconnaissance », à chaque tracé reçu, votre application devra retourner le geste reconnu, c'est-à-dire, le tracé contenu dans le dictionnaire qui se rapproche le plus du tracé reçu. Votre application devra alors envoyer un message sur le bus IVY sous la forme :

Geste nom=NomCommandeAssociéeAuGesteReconnu