

Projet long : reconnaissance de mouvement via Kinect

Le but est d'arriver à un logiciel reconnaissant les mouvements d'une personne faisant une présentation de manière à se passer de souris.

Matériel nécessaire : un Kinect, un vidéo projecteur et un ordinateur pour relier les deux.

Langage de programmation utilisé : C# ou C++.

I. Pourquoi avoir choisi ce projet ?

Nous avons choisi ce projet avant tout parce qu'il nous intéresse tous les deux : nous pensons que c'est essentiel pour un projet long, car cela permet de maintenir l'envie de contribuer au projet lorsque les difficultés s'accumulent.

Il a été souligné lors du premier cours que le but du projet était moins d'aboutir à un logiciel final impeccable que de s'impliquer dans le développement d'un projet sur le long terme. En ce sens, ce projet nous semble adapté : il nous permettra de découvrir des technologies et concepts nouveaux (C# et la reconnaissance de mouvement), tout en nous basant sur ce que nous savons déjà. Le premier cours a souligné la nécessité de sortir de notre zone de confort, et nous pensons que ce projet est un bon compromis : nous avons déjà les connaissances techniques pour appréhender un projet de ce genre, mais nous ne connaissons pas les langages de programmation en question et n'avons jamais travaillé sur de la reconnaissance de mouvement.

Même si nous n'avons jamais fait de C# ou de reconnaissance de mouvement, Microsoft met à notre disposition des tutoriels complets sur l'utilisation de Kinect et la reconnaissance de mouvement. Il existe également de nombreux tutoriels sur les langages C# et C++, le premier ayant l'avantage d'être proche de Java et le second du C.

II. Quels sont nos buts ?

Kinect et les logiciels associés construisent en temps réel un graphe représentant le squelette de la personne devant la caméra. La construction simple (sans traitement des données plus complexe) se fait à 30fps. Le logiciel est plutôt précis, mais a tout de même ses limites, notamment lorsque la personne se place trop de profil par rapport à la caméra. Les objectifs de cette année seraient donc :

- écrire un logiciel utilisant permettant de remplacer une souris pendant une présentation,

c'est-à-dire déplacer le curseur et cliquer.. C'est notre objectif prioritaire, mais cela ne nous prendra probablement pas toute l'année .

-écrire un petit logiciel pour calibrer facilement la caméra : régler l'angle d'inclinaison de la caméra, préciser si la personne qui parle est assise ou debout, calibrer la position de la personne par rapport à la surface de projection. Ces fonctionnalités existent déjà dans les logiciels fournis par Microsoft mais sont dispersées dans plusieurs logiciels. L'idée est donc de tout centraliser.

-écrire une ou deux améliorations au programme, comme diversifier les actions possibles (scroller, sélectionner, double-cliquer) par exemple.

III. Est-ce réalisable ?

Nous pensons que le projet est tout à fait réalisable sur la durée du projet long. Nous pensons suivre ce calendrier :

7/10 => 21/10 (deux semaines) : initiation à C# ou apprentissage accéléré de C++.

21/10 => 4/11 (deux semaines) : réalisation du programme de calibrage (prise en main du langage choisi)

4/11 => 13/01 (dix semaines) : réalisation programme de traçage de main (pour remplacer la souris)

13/01 => 27/01 (deux semaines) : tampon (retard de développement, partiels, problèmes inattendus)

27/01 => 10/03 (six semaines) : première amélioration

10/03 => 21/04 (six semaines) : seconde amélioration

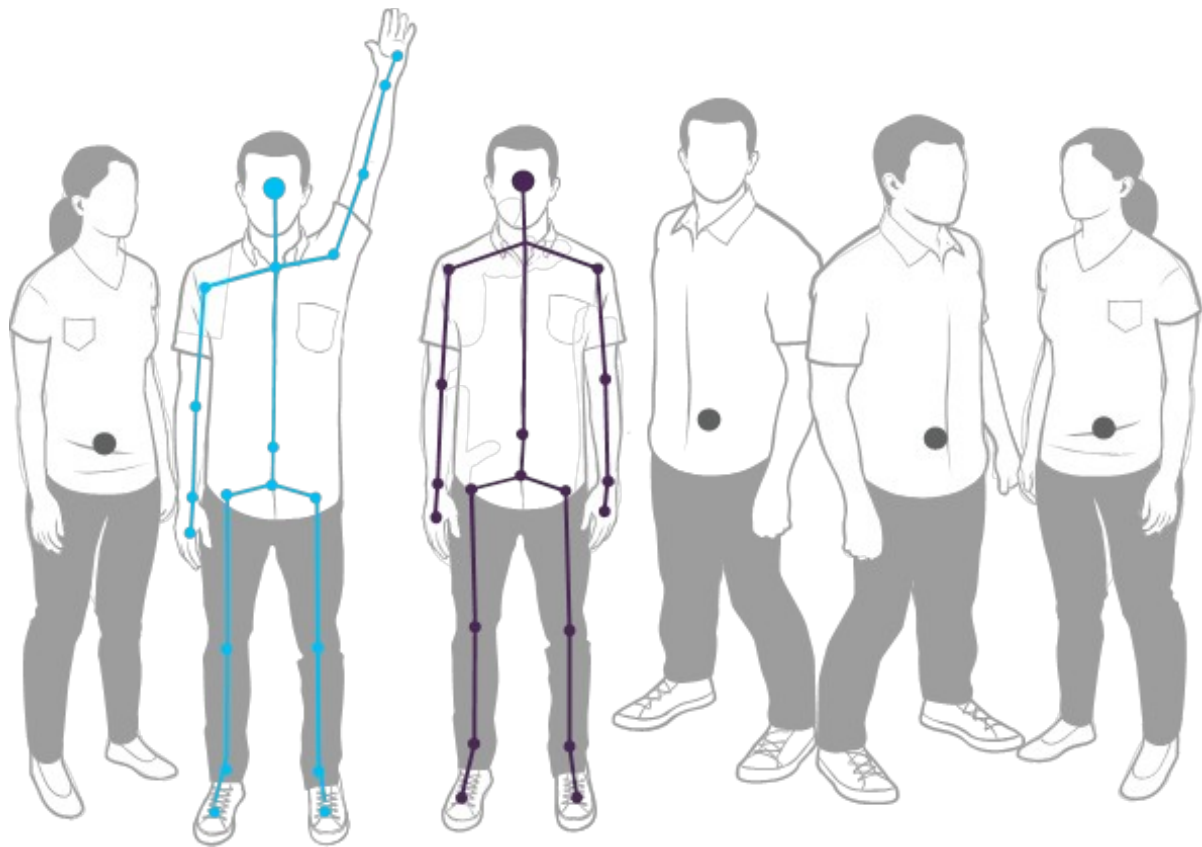
21/04 => rendu : retards de développement, debug, corrections de code

Présentation du SDK

Le SDK officiel est disponible sur Windows uniquement

(<https://msdn.microsoft.com/en-us/library/hh973074.aspx#ID4EY>)

Celui-ci s'occupe de la partie bas niveau de la reconnaissance de l'image, puis fournit sous la forme d'un graphe le squelette de la personne. Chaque noeud de ce graphe représente un noeud qui possède ses propres caractéristiques comme sa position dans l'espace.



Nous utiliserons certainement également OpenCV (<http://opencv.org>) de manière brève afin de réaliser la calibration entre le bureau et l'image projetée.