Devis technique de l'outil

Objectif de l'expérimentation

Le but de la recherche était d'identifier les fonctionnalités et les capacités de la Kinect pour son utilisation dans les jeux vidéo.

Nous avons donc ciblé les objectifs suivants en fonction du but donné :

- Détecter une personne
- Détecter deux personnes différentes en même temps sans conflit
- Détecter les mouvements de la main
- Bouger la palette en fonction de la position de la main
- Pouvoir détecter 2 personnes différentes en même temps
- Jouer à deux sur la même caméra
- Naviguer dans le menu avec la main
- Afficher la position de la main du joueur

Ces objectifs ont tous été remplis avec succès. Ils nous ont permis d'explorer les principes de reconnaissance de mouvements et de gestuelles.

Limites

Notre prototype garde la position des membres et articulations de notre corps pour contrôler un objet dans le jeu comme un curseur ou une palette. Il peut aussi reconnaitre des mouvements simples comme un balayement horizontal d'une main et un éloignement ou rapprochement vertical des deux mains. On peut aussi remettre les mouvements capturés sur un modèle et l'animer en temps réel.

Le prototype ne reconnait pas les mouvements plus complexes (tels qu'un rond, pousser ou tirer) et ne permet pas la détection de plusieurs mouvements en même temps. De plus, le balayement de la main est uniquement utilisé dans les menus et l'éloignement et rapprochement dans le jeu. Nous ne pouvons donc pas confirmer que les deux mouvements marcheraient bien conjointement. Finalement, le balayement vertical n'est pas supporté.

Éléments encore inconnus

Plusieurs éléments de la capture de mouvement avec la Kinect nous restent inconnus après les expérimentations. Les expérimentations n'ont pas permis d'exploiter les fonctionnalités

- La détection des doigts et de leurs gestuelles.
- La reconnaissance faciale.
- Toute fonctionnalité en dehors des limites du projet (reconnaissance vocale).
- L'attribution d'identifiants aux corps sur caméras. (La Kinect permet d'attribuer un ID de squelette pour conserver la priorité de détection. Par exemple, si on quitte la caméra et retourne dans le cadre, la Kinect nous reconnaît et nous garde en mémoire.
- Le near mode de la Kinect pour Windows n'a pas été exploité

Suivi des incertitudes

Une des premières grandes incertitudes que nous avions était le multijoueur, mais il fut très facile de l'implémenter. La Kinect le gère par elle-même et nous envoie toutes les données séparément.

Notre deuxième crainte était la capture de gestuelles qui fut relativement complexe. Nous nous sommes inspirés d'algorithmes déjà existants et des fonctionnalités offertes par le wrapper de Kinect pour Unity.

Nous voulions aussi assurer une marge d'erreur pour les gestuelles. Celle-ci fut un peu plus complexe à identifier afin de permettre un mouvement naturel, mais avec essai-erreur, nous avons trouvé ce que nous croyons être un juste milieu.

Référence

- Documentation officielle sur le SDK de Kinect for Windows ver.1.8

- Vidéo de la preuve de concept d’un Pong contrôlé avec la Kinect
- Fichiers pour l’intégration du SDK de Kinect for Windows ver.1.7 dans Unity
- Vidéo tutoriel de l’implémentation de la Kinect et de son SDK sur un poste de travail
- Exemples d’intégration de la Kinect dans Unity
- Exemple simple d’implémentation des mouvements de bras simples avec la Kinect dans une application
- Tutoriel d’implémentation de gestuelles avec la main pour la Kinect dans une application