TPI Kinect – Informations

# Projets et fonctionnalités réalisé(e)s jusqu’ici

## Les projets principaux

* PointTest : Dessiner à l’aide de Kinect
* PowerPointKinect : Add-in pour PowerPoint
  + Changer de slide à l’aide d’un mouvement (de droite à gauche, ou de gauche à droite), avec n’importe quelle main
  + Système de pointeur (indiquer sur le slide en cours un endroit en particulier à l’aide de mouvement)

## Les fonctionnalités de l’API vues

* Les différentes frames (traduit ici par « images ») provenant de Kinect
  + 30 fois par seconde, Kinect envoi des images à l’utilisateur. Plusieurs type d’image sont disponibles, par exemple l’image 2D en couleur, l’image infrarouge, ou encore une « image » qui contient une liste des corps (body) qui sont détectés par Kinect actuellement. Ce sont les données brutes.
* La classe fournie *KinectRegion*
  + Elle permet à une application Windows ou Windows Store de pouvoir être contrôlée avec Kinect à l’aide d’un minimum de ligne de code. Cette classe permet d’avoir une main à la place du curseur habituel
* La classe fournie *KinectCoreWindow*
  + Elle permet de gérer tout ce qui est de l’interaction entre l’utilisateur et l’application.
    - Par exemple, on peut connaitre la position des mains d’un utilisateur par rapport à son corps, en utilisant des valeurs x,y transposable facilement dans une application.
* La détection de mouvement/geste
  + Le logiciel KinectStudio
    - Permet de s’enregistrer devant la caméra faisant des gestes et d’en créer des clips
  + Le logiciel VisualGestureBuilder
    - Permet de créer, à partir de clip fait à partir de KinectStudio, une base de donnée de mouvement et de l’enregistrer dans un fichier .gdb. Ce fichier contiendra donc des gestes à détecter dans un programme
  + La détection de mouvement dans une application
    - Grâce aux classes disponibles dans *Microsoft.Kinect.VisualGestureBuilder*, on peut importer dans notre programme un fichier de base de données de mouvement .gbd et détecter ces gestes. Cela fonctionne de la même manière qu’un clic de souris : si Kinect détecte un mouvement, on effectue une action

## Les fonctionnalités non vues de l’API

* Fusion (testé mais pas utilisé)
  + Permet de scanner des objets pour en faire une représentation 3D
* Face - HDFace

## Les fonctionnalités vues ne faisant pas partie de l’API Kinect

* Développement d’application Windows Store
* Développement d’add-in pour PowerPoint
* Développement à l’aide de dll externes (*User32.dll*) pour interagir sur le curseur et le clavier. Au final, ce point était inutile, car non utilisé dans les projets effectués

## Les fonctionnalités à intégrer en plus ou à finir dans les projets

KinectPowerPoint :

* Résoudre le problème de déploiement automatique de certaines dll pour la détection des gestes. Ces dlls ne se déploient pas au bon endroit, pour le moment, la copie a été faite manuellement.
* Créer un programme d’installation ou un autre moyen qui ne nécessite pas l’installation du SDK Kinect
* Finir l’implémentation de la fonctionnalité de pointeur laser dans le PowerPoint

# Proposition de point à apparaitre dans le tutoriel

* Introduction à Kinect (Entreprise, version, fonctionnalités, projets réalisés par des tierces,…)
* Vue d’ensemble de l’API, sa structure, etc…
  + Utilisation de KinectStudio qui montre bien ce que Kinect voit
* Projet qui permet de dessiner avec Kinect
  + Petite introduction à la structure d’une application Windows Store (Application pour Windows 8)
  + Le code de base serait fournis à l’élève, ainsi, il n’aurait pas à développer l’application en elle-même, sa structure, etc… Car ce développement peut facilement devenir fastidieux. Il devrait à la place se concentrer uniquement sur le développement Kinect
* Projet de changement de slide PowerPoint avec les gestes
  + Introduction au développement d’Add-In pour PowerPoint, les méthodes utilisées, etc. De la même manière que pour le projet de dessin, on fournirait à l’élève le code de base.
  + Fournir, dans un premier temps, le fichier de base de données de mouvement .gbd à l’élève pour qu’il n’ait pas à le créer lui-même
  + Fournir aussi la classe *GestureDetector* créée par mes soins qui facilite la détection des gestes
  + Dans un deuxième temps, la classe *GestureDetector* serait développée par l’élève
  + A la fin de ce projet, une grande partie des fonctionnalités de l’API aurait été vue
* Tutoriel avancé, « pour aller plus loin… » (Kinect Avancé, suivant le temps disponible)
  + Création de fichier de base de données de mouvement .gbd à l’aide de KinectStudio et VisualStudioBuilder + introduction au type de mouvement *ProgressGesture* non utilisée dans les projets
  + Implémentation du curseur dans le PowerPoint