

# CONTROLE D'AVATAR PAR KINECT

## BENJAMIN BARBESANGE & BENOIT GARÇON

31/01/2015

## PRÉSENTATION

### 1. Description du projet

Dans le cadre de notre projet de première année à l'ISIMA, nous avons choisi de développer un programme de contrôle d'avatar 3D par une Kinect. Ce projet s'inscrit dans la continuité du cours de réalité virtuelle de deuxième année de Prep'ISIMA.

### 2. Objectif du projet

L'objectif sera donc de réussir à animer à l'écran un personnage (humanoïde) grâce aux mouvements de l'utilisateur numérisés par le capteur Kinect. Il s'agira donc dans un premier temps de développer un programme sous Processing capable de mettre en œuvre un tel mécanisme.

### 3. Ressources liées au projet

Voici une liste non exhaustive des différentes ressources nécessaires :

- Processing 2.2.1 : Environnement qui contient nativement des outils pour la 3D.
- Blender : utile pour le prétraitement des modèles 3D.
- SDK Kinect 2 et KinectPV2 0.7.2 : nécessaire au fonctionnement du capteur.
- Microsoft Kinect 2 : Ce capteur nous permettra l'acquisition des mouvements de l'utilisateur. En effet, grâce à ce dernier nous avons accès à pas moins de 25 points représentant des zones clés du squelette à 60fps. La liste :

▪ Épaule gauche	▪ Torse
▪ Coude gauche	▪ Cou
▪ Poignet gauche	▪ Tête
▪ Paume gauche	▪ Bassin
▪ Pouce gauche	▪ Hanche gauche
▪ Index gauche	▪ Genou gauche
▪ Épaule droite	▪ Cheville gauche
▪ Coude droit	▪ Pied gauche
▪ Poignet droit	▪ Hanche droite
▪ Paume droite	▪ Genou droit
▪ Pouce droit	▪ Cheville droite
▪ Index droit	▪ Pied droit
▪ Racine	
- Modèle 3D : nous aurons besoin de modèles compatibles avec les données acquises par le capteur, ainsi pour simplifier les choses nous donnerons au modèle le même squelette que celui acquis par le capteur. Ci-dessous, l'arbre représentatif du dit squelette

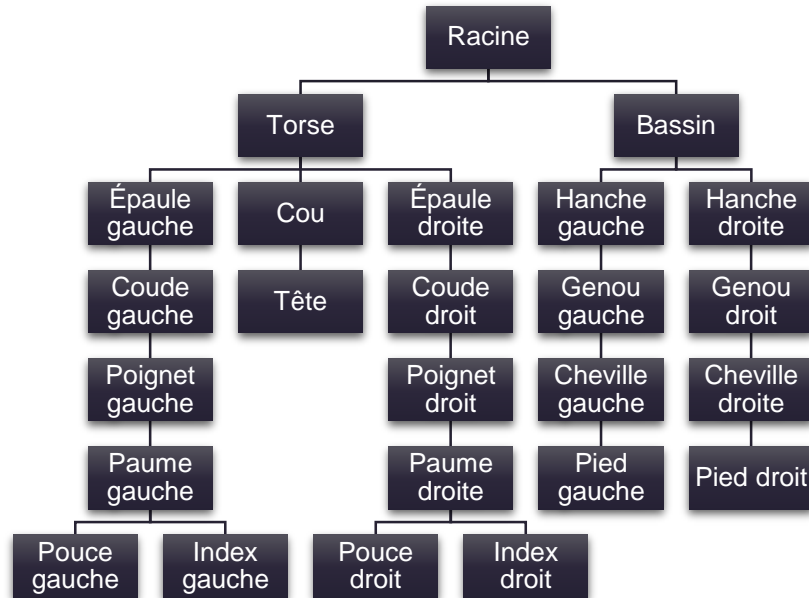


Figure 1 - Arbre des points du squelette

## MISE EN ŒUVRE

### 1. Plan de mise en œuvre

Le projet sera disponible à tout moment sur git à l'adresse suivante :

### 2. Algorithme principal

1. Acquisition des données
2. Normalisation des données
3. Calcul des matrices de transformation associées à chaque nœud
4. Application pondérée sur les vertices liées
5. Affichage du résultat
6. Retour à l'étape 1

### 3. Chronologie/planification principale

1. Préparation d'un modèle
2. Manipulation manuelle du modèle sous Processing
3. Elaboration du traitement principal
4. Ajout de fonctionnalités supplémentaires (débuggage, changement de modèle, ...)

### 4. Mots-clés

- Quaternion
- Interpolation
- Bone
- Vertex
- Rigging
- Skinning
- Weighting
- Animation squelettale
- Matrice de transformation