# Fi-Info5 – Bibliothèques de Développement Multimédia

## Projet 2019-2020 – Interaction avec la WebCam

## 1 Objectifs des séances

— Paramétrer l'environnement de développement pour utiliser les bibliothèque *OpenCV*.
— Savoir utiliser une fonction manipulant des images à partir d'un exemple de la documentation.
— Mettre en place une interaction avec l'utilisateur à partir de l'acquisition et du traitement des images issues d'une *WebCam*.
— Intégrer dans une même application la partie interaction et la partie affichage 3D.

## 2 Description de l'interaction

Il s'agit de mettre en place des fonctionnalités d'interaction entre le joueur et l'application par l'intermédiaire d'une *WebCam*. Dans l'application finale, le joueur contrôlera son propre déplacement dans le labyrinthe à partir de mouvements de rotation de sa tête. Ces mouvements seront visualisés dans une fenêtre, intégrée à l'interface graphique, affichant une partie du champ de la caméra. Pour la mise au point de ces fonctions d'interaction, on pourra commencer par créer une application comprenant une zone graphique contenant un objet symbolisant le joueur sous la forme d'un petit disque avec un segment indiquant la direction de déplacement (cf. figure 1). Cet objet se déplacera dans la zone graphique de la même façon que le joueur sur la carte du labyrinthe. En position neutre, le visage est de face par rapport à la caméra et peut se situer à n'importe quelle position dans le champ. Cette position neutre devra être déterminée automatiquement par le programme lorsque le visage reste de face et quasiment immobile pendant quelques instants. Il pourra revenir en position neutre à n'importe quel moment pendant le jeu. A partir de cette position neutre, deux types de mouvements seront détectées :

1. la tête tourne légèrement à droite ou à gauche : l'objet tourne sur lui-même vers la droite ou vers la gauche tant que la tête reste tournée, il s'arrête de tourner dès que la tête revient en position neutre.
2. la tête est légèrement levée ou baissée : l'objet se déplace en avant ou en arrière dans la direction indiquée par l'objet tant que la tête reste inclinée, il s'arrête dès que la tête revient en position neutre.

Il faudra veiller à obtenir une interaction la plus fluide et fiable possible pour obtenir une bonne expérience utilisateur dans l'application finale.

## 3 Etapes à franchir

1. Installer la bibliothèque *OpenCV* dans son environnement de développement.
2. Mettre en place une application pour la mise au point de l'interface. Cette application aura une zone de captation du geste et une zone de dessin de l'objet en mouvement (ici un petit disque pour la mise au point). On pourra partir de l'exemple *TestWebCamQt* fourni.
3. Etudier la fonction *detectMultiScale* et son application à la détection de visage dans l'exemple *TestDetectMultiScale* donné sur Mootse et sur la page de documentation d'*OpenCV*. La détection de visage sera utilisée pour déterminer la position neutre du visage.
4. Etudier la fonction *matchTemplate* dans l'exemple *TestDetectMotion* donné sur Mootse et sur la page de documentation d'*OpenCV*. Cette fonction sera utilisée pour détecter les petits mouvements de rotation ou d'inclinaison de la tête.

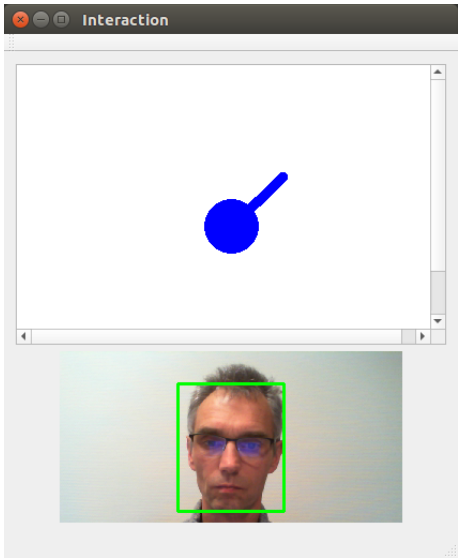


FIGURE 1 – Interface pour la mise au point de l'interaction

5. Ajouter la détection des mouvements de la tête et la visualisation du mouvement de l'objet à votre application. On pourra passer par les sous-étapes suivantes :
   i. Détection de la position neutre (on pourra par exemple attendre plusieurs détections successives du visage sans qu'il y ait de mouvements).
   ii. Détection des rotations droites-gauches et des inclinaisons haut-bas de la tête. Visualisation de la rotation et du déplacement de l'objet en fonction de l'orientation de la tête.
   iii. Optimisation des paramètres pour augmenter la fiabilité et la fluidité. On utilisera pour cela les fonctions de multi-threading du C++11. Vous pouvez vous reporter à l'exemple *TestMultiThreading* donné sur Mootse et sur les pages de documentation de *cplusplus-reference*.