

M o d u l e d ' I H M

E t u d e d e f a i s a b i l i t é d e n a v i g a t i o n g e s t u e l l e

Patrick BENOIT - François POYER

Christophe SOUM - Baptiste VALLIERE



CSTB - Encadré par A.M.Dery et D.Lingrand

Introduction

Le sujet proposé pour ce projet de fin de semestre dans le cadre du module d'interaction homme-machine était relativement libre. La seule contrainte était de mettre en valeur la salle d'immersion du Centre Scientifique et Technique du Bâtiment (CSTB). Amateurs de science-fiction et le film ayant été montré en exemple lors d'un des premiers cours d'IHM, nous souhaitons reproduire une interface à la «Minority Report». Après une séance de démonstration des fonctionnalités de la salle, nous avons trouvé qu'une application de navigation 3D serait la plus adaptée pour mettre la salle en valeur. Après une présentation du CSTB et de sa salle, nous allons décrire étape par étape la réalisation du prototype de cette application.

Présentation du CSTB

En attente

Partie I - Modèle utilisateur :

I -1 : Qui sont les utilisateurs :

Les utilisateurs sont des personnes novices en informatique et en navigation 3D. En effet, les utilisateurs de la salle ne sont pas familiers avec celle-ci. De plus, même si ils travaillent dans le bâtiment, la navigation dans un univers de synthèse n'est pas quelque chose à laquelle ils sont habitués.

De ce fait, pour faciliter l'utilisation et la compréhension du logiciel, l'interface par mouvements de la main doit se faire de la manière la plus simple et la plus intuitive. Pour finir, l'utilisateur n'a pas l'habitude d'utiliser du matériel de haute précision, il n'est donc pas préférable de lui imposer d'utiliser une structure lourde ou fragile. Il faudra donc utiliser un matériel léger, souple et bon marché.

I - 2 : Modèle de navigation :

Il existe 2 modèles de déplacement et de vue 3D :

- *vue centrée (point de vue = objectif en vue) : type modeleur 3D*

Ce qui est au centre de l'attention de l'utilisateur est au centre du point de vue et ne peut être quitté des yeux. La vue est «en orbite» autour d'un point central devant l'objectif de la caméra virtuelle. Avantage majeur : utilisation aisée. Inconvénient majeur : forte restriction dans les déplacements (pas de translations hormis zoom avant ou arrière sur le point/l'objet visé).

- *vue non centrée (point de vue = visée caméra) : type jeux FPS*

L'utilisateur dirige son point de vue comme une caméra subjective. Le point de vue est en rotation autour d'un point situé derrière la caméra virtuelle (la «tête» de l'avatar virtuel qui «tient la caméra»). Avantage majeur : liberté totale des déplacements dans les limites de l'univers virtuel. Inconvénient majeur : peu intuitif, car nécessite l'utilisation de deux entrées simultanées : translation (jambes) et rotation (tête), ce qui peut «perdre» l'utilisateur.

Pour réaliser cette interface, nous désirons offrir à l'utilisateur la liberté de ses déplacements, nous allons donc choisir une vue non centrée.

I - 3 : Etude de l'existant :

I - 3 - 1 : Les périphériques d'entrée :

La navigation et l'interaction dans des univers virtuels avec ou non immersion de l'utilisateur est un problème connu et documenté. Au fur et à mesure de l'évolution du matériel, différents périphériques d'entrée sont apparus. Nous allons en dresser une liste non exhaustive :

- Clavier + souris :

La combinaison de ces deux périphériques très classiques est surtout répandue dans le domaine des jeux vidéos où il est nécessaire de naviguer. Cela reste cependant peu intuitif et demande un certain entraînement pour être maîtrisé. Il est en effet demandé à l'utilisateur d'effectuer des mouvements très différents des deux mains à la fois. Une configuration de la souris (inversion ou non de l'axe vertical) peut être proposée pour s'adapter aux habitudes de l'utilisateur.



La combinaison Clavier + Souris est la plus utilisée malgré sa difficulté de prise en main, pour des raisons de prix et de disponibilité de matériel.

- Joysticks :

Un périphérique très classique lui aussi lorsqu'il s'agit de navigation. Plus intuitif que la combinaison Clavier + Souris, car l'utilisation d'une seule main à la fois est requise, sa limite principale est le faible nombre d'opérations proposées : peu de boutons entraîne un maniement simple mais moins de possibilités, ce qui peut être compensé par l'ajout de boutons supplémentaires. Mais dans ce cas le problème de l'utilisation des deux mains (ou de différents doigts) pour des tâches radicalement différentes et simultanées se pose à nouveau.



Un joystick «manche à balai» classique. L'utilisation du pouce droit en même temps que celle de l'index droit ou de l'ensemble de la main droite peut être compliquée.

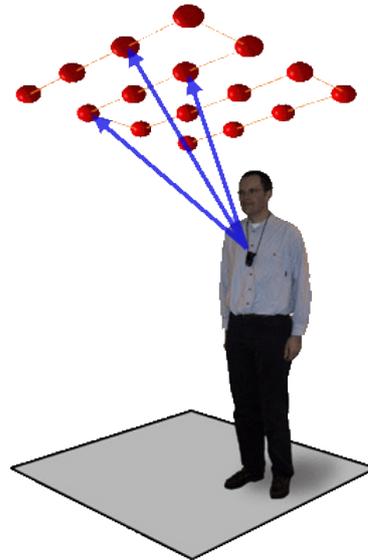
- Souris 3D / joysticks 3D :

Conçus sur la base d'une souris «classique» ou d'un joystick, certains périphériques proposent d'interagir directement en 3D. En contrepartie, comme ils permettent un maximum d'opérations tout en conservant un nombre limité de boutons/manettes/leviers, ils sont en général très peu intuitifs.



Divers modèles de joysticks et souris 3D.

Dans les cas où leur utilisation est simple, ils demandent souvent un matériel coûteux.

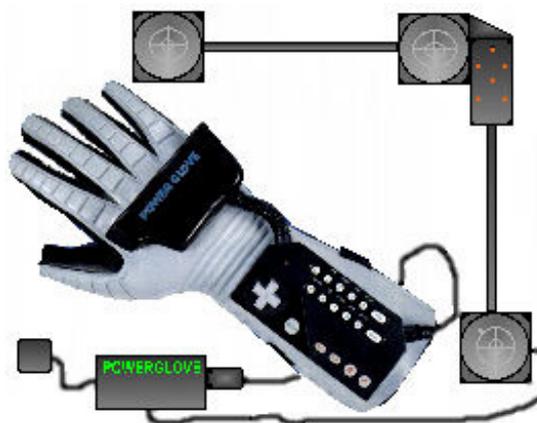


*Repérage de la position d'un souris 3D portée au cou (système Bat)
par une série de détecteurs placés au plafond.*

De façon générale, leur utilisation est très spécifique. En effet, il est assez difficile de les utiliser dans un contexte 2D classique. Cette caractéristique et leur prix souvent assez élevé fait qu'ils ne sont que peu répandus.

- Power gloves / Data gloves :

Les gants à capteurs sont rarement utilisés pour une interaction avec un contexte 3D. Ils ont l'énorme avantage d'être très intuitifs d'utilisation : il suffit de bouger sa main pour les faire fonctionner. Les mouvements, souvent simples comme ouvrir - fermer la main, déclenchent des capteurs qui envoient les instructions au système. Nintendo a tenté de distribuer largement des gants de ce type (les Power Gloves), mais leur prix prohibitif pour l'époque (500 Frcs) les a empêché de connaître un véritable succès. De plus ces gants ne permettent que peu de mouvements comparés aux besoins d'une application en contexte 3D et sont assez encombrants.



Le Power Glove Nintendo.

D'autres gants à capteur disposent de suffisamment de degrés de liberté pour une navigation 3D (6 degrés: 3 translations, 3 rotations), ils sont légers et peu encombrants car sans fil tout comme notre solution. Mais leurs prix sont très élevés car ils atteignent les 995\$ pour la version sans fil (Fifth Dimension Tech).



Fifth Dimension Tech s'est spécialisé dans l'interaction par gants.

- *Périphérique atypiques (vélo, tapis roulant, ...)* :

D'autres périphériques d'interaction en 3D existent. Ils sont souvent développés au cours d'expérimentations, de tests ou dans un but précis, et sont rarement adaptables à un autre contexte. Originaux mais coûteux, ils demandent parfois une certaine forme physique... et ne permettent souvent que des déplacements en 2D dans l'univers 3D.



Le vélo pour une ballade dans une ville virtuelle.

I - 3 - 2 : Avantages et inconvénients des périphériques :

De nombreuses recherches ont été menées sur le sujet de l'interaction avec un univers 3D virtuel, et en particulier sur les avantages et inconvénients des différents types de périphériques utilisés pour une action donnée. Nous allons vous présenter ici une synthèse des principaux points dégagés.

- Pointage :

Ware [25] a montré que le placement approximatif d'objets en trois dimensions devenait trivial avec leur capteur de position 3D The Bat, comparé à la souris.

Une étude de Driver [6] qui compare un trackball 3D, une souris et un capteur de position 3D dans une tâche de pointage d'objet, révèle la supériorité du capteur 3D. Les utilisateurs font moins d'erreurs de perception en trois dimensions grâce au retour d'information kinesthésique.

Zhai [28] montre que les périphériques isotoniques (type joystick analogique) sont les plus intuitifs et les plus rapides à maîtriser.

Hinckley [10] suggère à partir de l'analyse des travaux d'Yves Guiard que les déplacements d'un périphérique dans un espace fixé (sans configuration possible) diminuent l'efficacité de l'utilisateur.

Les capteurs de position 3D sont donc intéressants pour le pointage d'objet, mais cela pose des problèmes de perception des mouvements entre l'espace réel et l'espace virtuel [23].

L'utilisation de gants nous permettrait d'avoir à la fois les avantages d'un capteur 3D, c'est-à-dire les informations kinesthésiques qui donnent une sensation de contrôle à l'utilisateur, et d'un périphérique isotonique, à savoir la prise en main rapide car intuitive et la facilité de conversion des mouvements entre espace réel et virtuel.

- Sélection :

Yoshimura [26] justifie l'utilisation de la manipulation directe en 3D en reprenant la définition de Schneiderman [20]. Il utilise pour son système d'interaction 3D un pointeur 3D se terminant par un rayon qui traverse les objets dans l'axe du pointeur (technique du ray casting). On utilise dans ce cas l'orientation du capteur 3D plutôt que sa position. La méthode classique consistant à sélectionner l'objet une fois que le pointeur (i.e. le capteur de position 3D) est directement positionné dessus.

Cette technique étant malgré tout relativement lourde à mettre en place, nous ne pourrions l'utiliser. Il nous faudra donc faire en sorte que l'utilisateur n'ait aucun doute sur l'endroit où il pointe. La présence d'un curseur semble la solution la plus utilisée car la plus intuitive (retour visuel).

- Interaction à deux mains :

Kabbash [13] met en avant les applications pour lesquelles l'utilisation des deux mains est positive, et n'apporte pas une charge cognitive supplémentaire pour l'utilisateur. En effet, il a constaté que l'utilisation des deux mains peut être pire que celle d'une seule si les applications assignent une tâche indépendante à chaque main. Cependant, cela permet aussi dans certains cas de s'adapter beaucoup plus rapidement, d'aller plus vite à l'information ou de manipuler plus facilement l'interface.

Dans notre cas, l'utilisation en parallèle des deux mains risque effectivement de compliquer fortement l'utilisation. Il faut donc laisser la possibilité d'un choix entre l'utilisation d'une seule main - tout doit pouvoir être fait avec une seule main, même si c'est un peu plus long - ou des deux mains - plus compliqué mais plus rapide.

En conclusion, l'utilisation d'un capteur de position 3D gestuel, offre à la fois une bonne facilité de prise en main et une précision suffisante pour l'application visée. Il faut cependant laisser à l'utilisateur la possibilité de ne se servir que d'une seule de ses mains à la fois.

I - 3 - 3 : Communication gestuelle :

Le choix d'une interaction par mouvement des mains étant fait, il nous faut maintenant décider des gestes que l'utilisateur effectuera pour interagir avec l'univers virtuel.

Comme nous voulons une interface simple et facile à utiliser, nous avons pensé à consulter des études sur les langages gestuels : Le langage des signes comporte un vocabulaire très vaste pour toutes les situations de la vie quotidienne ou non.

De part sa grande expressivité, les signes ne sont pas vraiment intuitifs car ils doivent pouvoir être facilement différenciés. Dans notre cadre d'utilisation, un vocabulaire plus restreint est suffisant, mais il doit être intuitif.

De ce point de vue, le document le plus pertinent que nous ayons trouvé est la thèse de M. Lebourque [1] dont la première partie traite des gestes intuitifs. Mais nous n'avons trouvé aucune application reconnaissant des mouvements de la main grâce à une caméra.

Après quelques recherches nous avons constaté que le langage des signes n'est pas homogène au niveau international. Or, les utilisateurs de la salle peuvent venir de l'étranger. L'utilisation du langage des signes pourrait donc induire certains utilisateurs en erreur et est donc à proscrire.

Nous allons donc devoir 'inventer' notre propre code de signes afin qu'il soit le plus naturel et le plus intuitif possible, tout en faisant en sorte qu'il apporte le moins d'ambiguïtés possible.

Partie II - Modèle de conception

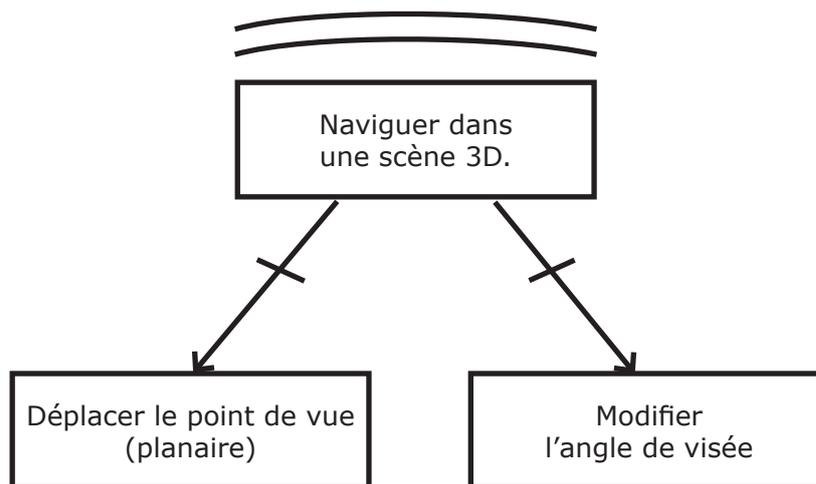
II.1 : Modèle HTA

II.1.1 : Définition des tâches

Afin de réaliser une application de navigation, nous devons permettre de voir la scène sous l'angle désiré. Pour cela, les trois tâches principales sont :

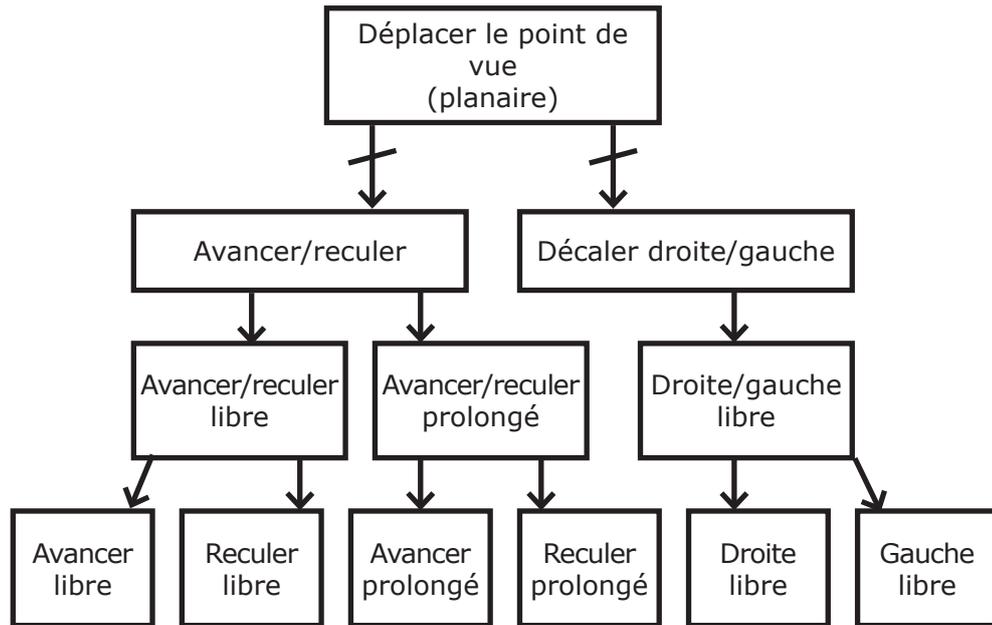
- Déplacer le point de vue
- Changer l'angle de vue
- Permettre une interaction minimum avec le monde virtuel.

Nous allons décrire maintenant en détail chacune des tâches.



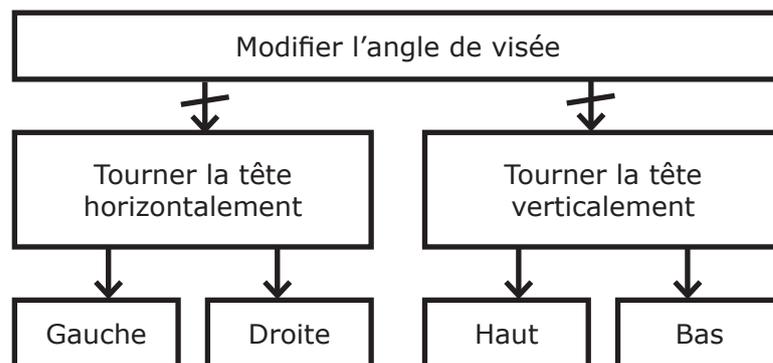
Naviguer dans une scène 3D (décomposé)

Cette première tâche principale permet à l'utilisateur de 'visiter' le monde virtuel. Il est important qu'il puisse à tout moment changer son angle de vision sans s'arrêter de se déplacer tout comme un promeneur peut tourner la tête sans s'arrêter de marcher. L'utilisateur doit donc avoir le choix entre un déplacement lié à son angle de visée (type jeu First Person Shooter) et un déplacement indépendant de sa ligne de visée (type tourelle). Dans les deux cas, les tâches restent les mêmes, seuls les repères auxquels sont appliquées les transformations varient.



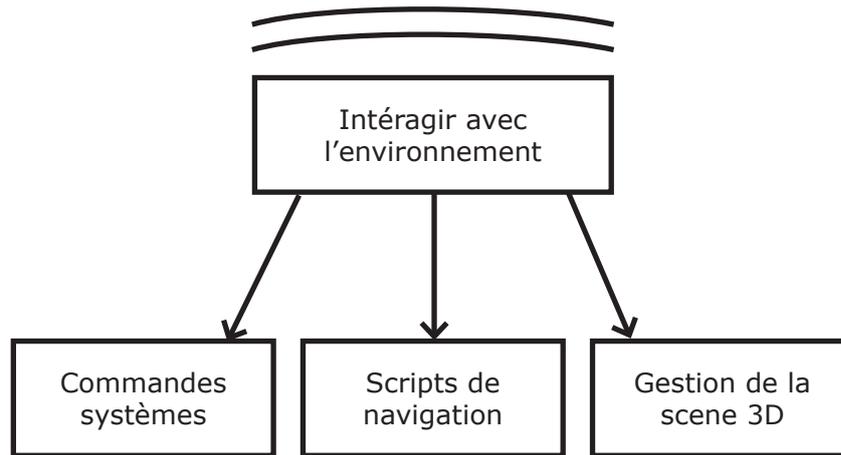
Se déplacer dans une scène 3D.

En ce qui concerne la première sous-tâche, le déplacement du point de vue, afin de limiter les efforts de l'utilisateur (essayer de garder le moins possible les mains en l'air car c'est un geste rapidement épuisant) l'idée de déplacement prolongé permet à l'utilisateur de prolonger jusqu'à une nouvelle commande le dernier mouvement effectué.



Panoramique dans la scène 3D.

La deuxième sous-tâche permet à l'utilisateur de modifier le vecteur de visée (simule les rotations de la tête). Il peut ainsi regarder librement le monde virtuel autour de lui.



Interagir avec l'environnement

Bien que n'étant pas nécessaire directement à la navigation, cette tâche est importante car elle permet à l'utilisateur de ne pas rester passif dans l'univers virtuel qui l'entoure. En effet, toutes les actions décrites précédemment concernaient exclusivement la navigation: il fallait fournir à l'utilisateur la possibilité de réaliser des opérations connexes et indépendantes. Comme ces tâches peuvent potentiellement être répétées assez souvent, il est nécessaire qu'elle soit simple à effectuée et pas trop fatigante ou inhabituelle (muscles peu habitués à travailler longtemps).

II.2 : Modèle UAN

Dans cette partie, nous allons décrire les tâches des arbres précédents. Nous allons nous limiter aux tâches les plus importantes à savoir : naviguer dans une scène en 3 dimensions et effectuer des commandes d'interaction avec le système ou la scène.

Dans toute cette partie, nous utiliserons les définitions suivantes (inversibles pour les gauchers) :

- MD et MG, respectivement la main droite et la main gauche de l'utilisateur.
- ←, ↑, → et ↓ : directions de déplacement.
- B mouvement de blocage (main fermée).

Nous allons détailler les modèles UAN des tâches précédentes en prenant en compte un mode de déplacement type 'FPS'. Cependant, comme dans le cas des modèles HTA, les modèles UAN de l'autre mode de déplacement sont identiques à ceci près que les notions de avant, arrière, gauche et droite ne sont plus celles implicitement liées à la vue, mais aux direction cardinales du monde virtuel (par ex : avant = nord, ...).

II.2.1 : Modèle UAN de la tâche 'naviguer dans une scène 3D'

Tâche : déplacer le point de vue		
<i>User actions</i>	<i>Interface feedback</i>	<i>Interface state</i>
MG ←	le point de vue de l'utilisateur se déplace vers la gauche	l'avatar de l'utilisateur se déplace vers la gauche dans l'univers
MG ↑	le point de vue de l'utilisateur se déplace vers l'avant	l'avatar de l'utilisateur se déplace vers l'avant dans l'univers
MG →	le point de vue de l'utilisateur se déplace vers la droite	l'avatar de l'utilisateur se déplace vers la droite dans l'univers
MG ↓	le point de vue de l'utilisateur se déplace vers l'arrière	l'avatar de l'utilisateur se déplace vers l'arrière dans l'univers
MG B	le point de vue poursuit le dernier mouvement Avant/arrière effectué	l'avatar de l'utilisateur continue le dernier déplacement Avant/arrière entamé
MG →B	le point de vue est à nouveau sous contrôle de l'utilisateur	l'avatar de l'utilisateur répond à nouveau aux ordres de déplacement qui lui sont donnés

Tâche : modifier l'angle de visée		
<i>User actions</i>	<i>Interface feedback</i>	<i>Interface state</i>
MD ←	l'angle de visée est dirigé plus à gauche	l'avatar de l'utilisateur tourne la tête vers la gauche
MD ↑	l'angle de visée est dirigé plus en haut	l'avatar de l'utilisateur lève la tête
MD →	l'angle de visée est dirigé plus à droite	l'avatar de l'utilisateur tourne la tête vers la droite
MD ↓	l'angle de visée est dirigé plus en bas	l'avatar de l'utilisateur baisse la tête

II.2.3 : Modèle UAN de la tâche 'Interaction avec l'environnement'

Afin d'éviter les conflits avec les commandes de navigation, nous avons décidé que les commandes utilisateurs seront détectées dans une portion dédiée de l'espace de capture. Afin de rester le plus général possible, l'action sur les commandes s'effectue par détection d'une position de main prédéfinie et redéfinissable. Ces stimuli sont ensuite reportés à l'environnement qui les traite selon sa configuration (actions scriptables).

Dans la suite, nous utiliserons les définitions suivantes :

- BC définit la boîte de commande
- R définit un repère de diode
- C définit une commande (nombre de R et position relative).

Tâche : Utiliser une commande		
<i>User actions</i>	<i>Interface feedback</i>	<i>Interface state</i>
~>M[BC]	~R[BC]	Détection de la position des R dans BC
C	Retour du numéro de C	La commande est traitée.

Partie III - Réalisation du prototype

III.1 Objectifs:

Etant donné la spécificité de notre projet, nous avons décidé de réaliser une application concrète afin d'évaluer la faisabilité d'un tel dispositif.

L'estimation de la qualité de la prise en main se base sur l'ergonomie du périphérique. Ergonomie dépendant de multiples paramètres, donc impossible à évaluer sous forme de magicien d'Oz.

Cependant nous devons trouver une manière pour éviter une surenchère de développement du point de vue vision. De plus, nous avons pour contrainte le CSTB comme cadre de test: le programme devant se greffer sur leur système, nous n'avons pas le droit à l'erreur (n'ayant pas accès à la salle).

Nous avons adaptés les propositions précédemment effectuées aux contraintes résultantes de l'interfaçage avec la machine du CSTB. Nous ne pouvions ni diversifier les modes de déplacement - le mode «tourelle» a notamment été abandonné - ni implémenter le contrôle du pointeur (souris), et l'interaction avec la scène (sélection). En effet, nous utilisons le flux de communication dédié à l'origine pour la navigation par joystick (informations d'axes et boutons uniquement).

III.2 Implémentation:

III.2.1 Choix techniques:

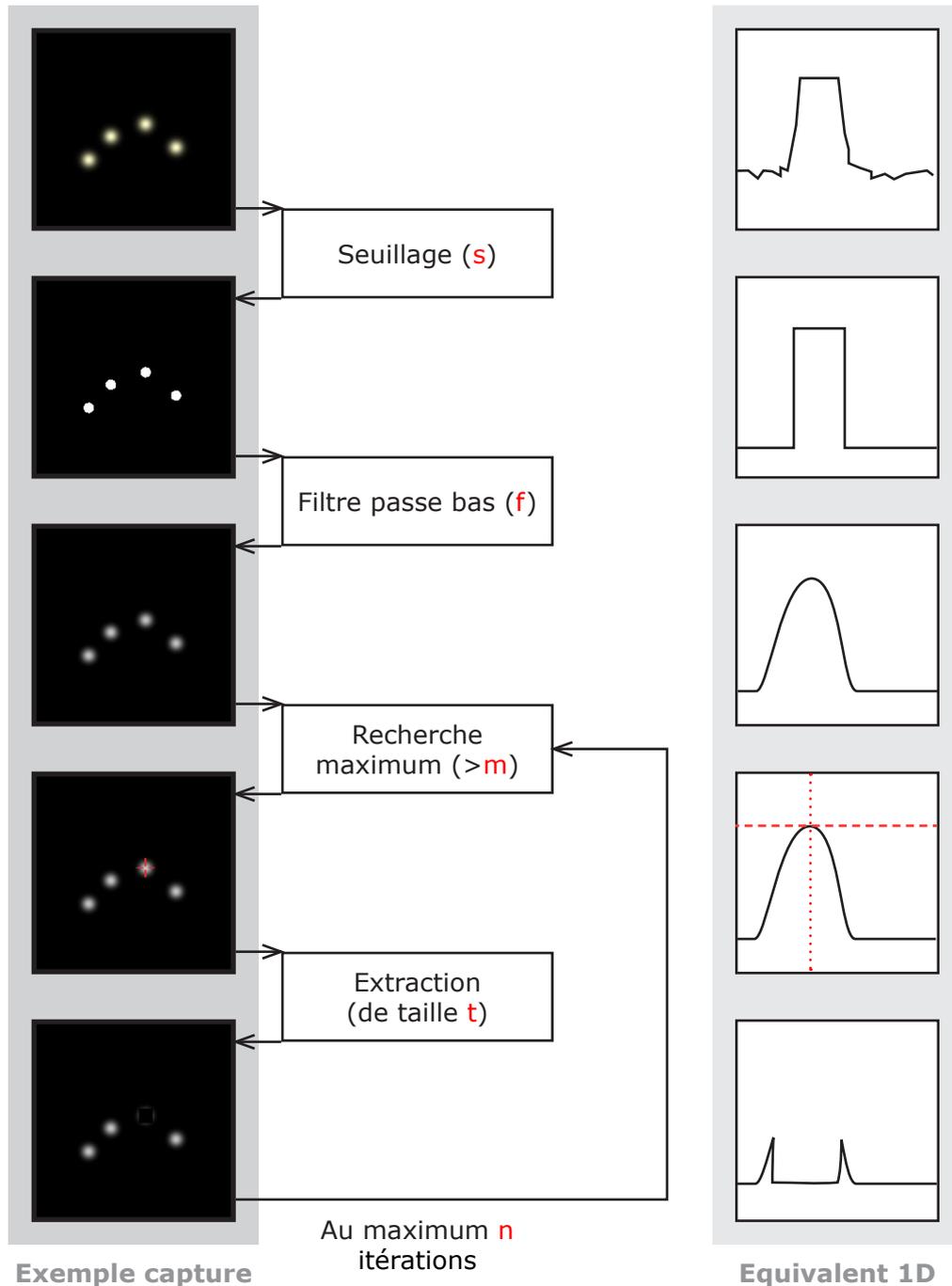
Nous avons développé notre application sous Windows afin de bénéficier de la librairie DirectShow pour la gestion de la capture vidéo à partir de la webcam (utilisation du code du projet BreakBit [3]).

De plus nous utilisons la librairie graphique QT, ce qui nous a facilité la mise en place de l'interface, des communications, de l'interfaçage avec OpenGL, et du dessin des éléments de l'interface superposés à l'image capturée.

III.2.2 Traitement vision:

Afin de réaliser la détection des diodes, nous avons élaboré un algorithme de traitement en plusieurs passes. La détection devait répondre à divers critères: notamment une caractérisation spatiale précise des diodes, étant donné le ratio important entre la résolution de l'écran du CSTB et la résolution de la webcam.

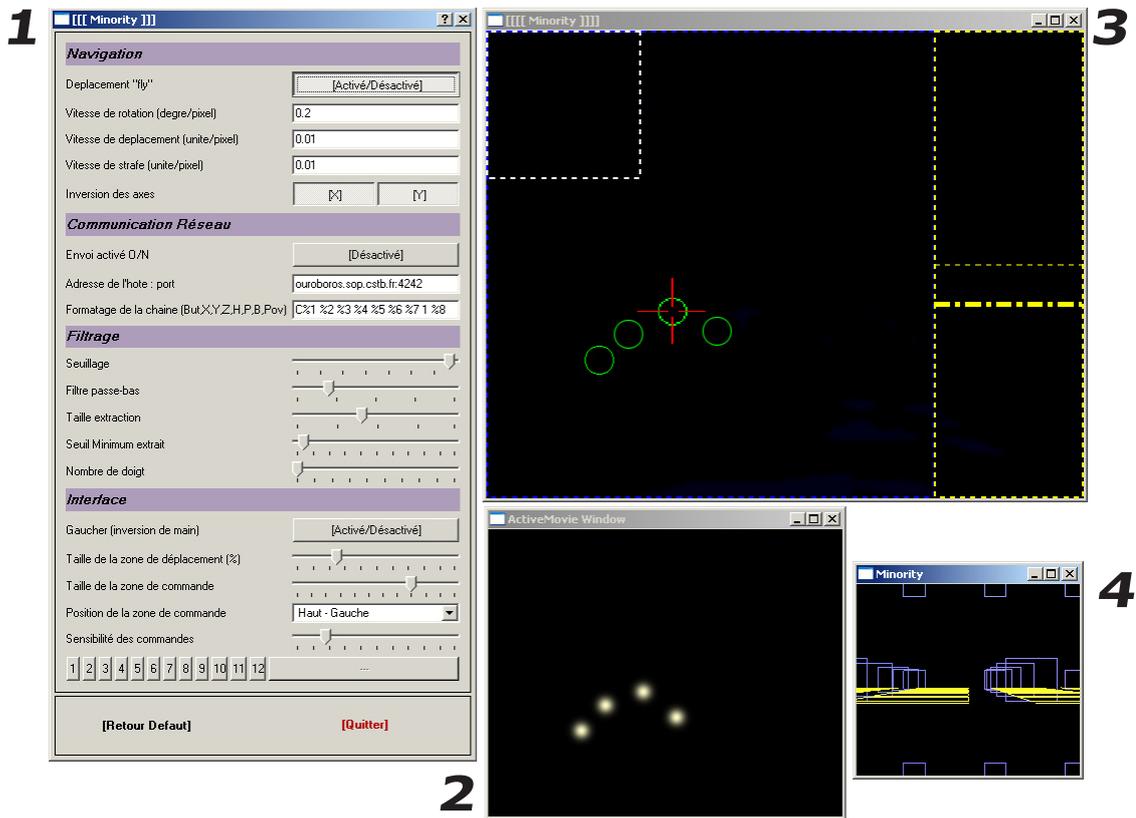
C'est pourquoi, nous sommes passés par les traitements suivants:



III.2.3 Interface:

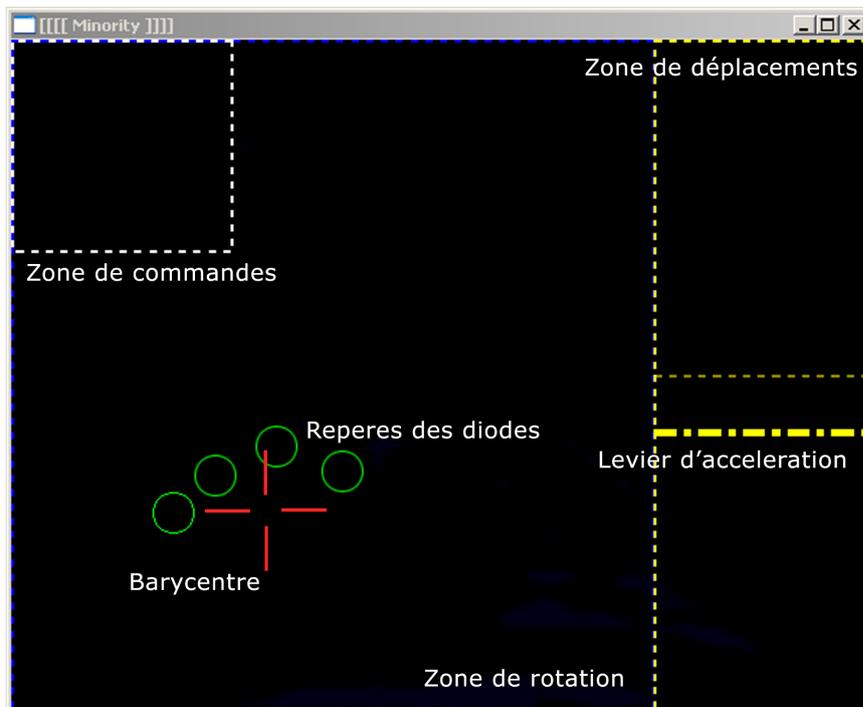
L'interface se compose:

- 1- D'une fenêtre permettant de contrôler les principaux paramètres de la navigation, du traitement, de la communication réseau et de la disposition de l'interface.
- 2- D'une fenêtre de retour vidéo.
- 3- D'une fenêtre de retour commande.
- 4- Et enfin, une fenêtre de visualisation OpenGL permettant de simuler la navigation dans une scène 3D.



Les différentes fenêtres composant l'interface.

La fenêtre de retour commande est divisée ainsi:



Partie IV : Evaluation du prototype

L'évaluation du prototype s'est faite le 29 mars au CSTB. Une première partie de mise au point a été nécessaire pour nous permettre de régler quelques paramètres de l'application. Etant donné que l'application a été bien conçue, tout était entièrement configurable à la dernière minute pour coller le mieux possible à l'application existante de navigation en 3D du CSTB.

Après cette phase de configuration qui fut assez courte, Christophe a testé la navigation dans un univers de petite taille, puis s'est essayé sur une carte représentant le département en 3D. Nous en avons profité pour effectuer différents réglages sur la sensibilité des mouvements, mais aussi des seuils de détection afin de ne repérer que les diodes dans l'image. Malgré quelques parasites, le système était prêt pour la phase de test avec des utilisateurs ne connaissant pas le module.

Nous disposions de cinq 'cobayes', que nous avons tout d'abord observé pendant qu'ils testaient l'application, puis nous leur avons posé toute une série de questions. Voici la liste des observations et des questions et les conclusions que nous en avons tiré:

IV.1 Questions posées à l'utilisateur:

Q1 - Avez-vous trouvé facile l'utilisation de ce périphérique ?

R1 - L'utilisation du périphérique a paru assez intuitive, malgré un temps d'adaptation plus ou moins important.

Q2 - Avez-vous trouvé les mouvements fatigants musculairement ?

R2 - Les utilisateurs n'ont pas trouvé le test fatigant, mais il était trop court pour permettre d'en juger.

Q3 - Avez-vous trouvé que les mouvements gérant le déplacement et le point de vue étaient imprécis ?

R3 - Les utilisateurs ont trouvé que le déplacement était imprécis dans le cas où ils utilisaient cinq doigts mais fluide lorsqu'ils n'en utilisaient qu'un.

Q4 - Avez-vous trouvé gênant l'usage simultané des deux mains ?

R4 - L'usage simultané des deux mains a été un peu difficile au départ pour la plupart, mais ils pensent s'être adapté assez rapidement.

Q5 - Auriez-vous préféré une inversion des fonctions de la main gauche et main droite ?

R5 - Non dans l'ensemble. Le module permet quand même cette inversion.

Q6 - Avez-vous trouvé les mouvements naturels, intuitifs ? Pensez-vous qu'il soit possible de s'y habituer rapidement ? A l'usage ?

R6 - Les mouvements leur ont paru assez naturels. La rotation a paru moins facile certain d'entre eux qui aurait un autre mode (jamais le même).

Q7 - Est-ce que l'usage des gants vous a gêné/déplu (lumières, matière, ...) ?

R7 - Aucune remarque négative n'a été faite concernant l'usage des gants.

Q8 - Est-ce que vous avez trouvé un avantage par rapport aux autres périphériques ?

R8 - Le sentiment principal des utilisateurs a été une plus grande sensation d'immersion due au retour d'information kinesthésique.

L'aspect ludique et démonstratif a été également évoqué.

Plus comparativement le module a paru équivalent à un joystick (pour les utilisateurs habitués).

Q9 - Autres remarques ?

R9 - Aucune remarque générale n'est ressortie, seulement des propositions diverses comme l'utilisation de leviers pour gérer les rotations ou l'augmentation de l'immersion par le reste du corps.

Alain Giboin nous a signalé qu'il aurait été préférable d'ajouter les deux tests suivants :

- suivi d'un trajet indiqué oralement,
- recherche d'une zone précise,

afin d'évaluer plus précisément l'application.

Vu que l'utilisation d'un seul doigt par main a semblé préférable à certains utilisateurs, l'idée d'une simplification du support des diodes a été proposée. nous pourrions en effet utiliser deux baguettes pourvues d'interrupteurs au bout desquels serait fixée une diode unique.

IV.2 Questions posées aux observateurs:

Q1 - Avez-vous trouvé la navigation fatigante visuellement (maux de tête, nausées, ...) ?

R1 - La navigation a été trouvée assez fatigante à cause des tremblements et du manque de fluidité générée par l'utilisation de plusieurs doigts.

Q2 - Avez-vous trouvé la navigation fluide ?

R2 - Lorsque le manipulateur n'utilise qu'un doigt pour gérer la rotation, la navigation est fluide. Dans les autres cas elle est beaucoup plus saccadée.

Q3 - Est-ce que la présence des lumières vous a gêné/déplu ?

R3 - Globalement, les lumières n'ont gêné personne.

Q4 - Autres remarques ?

R4 - L'aspect ludique a été à nouveau souligné.

IV.3 Observations faites:

Q1 - Fatigue musculaire (l'utilisateur baisse les bras, tremble) :

R1 - Malgré les réponses à la question concernant ce point, nous avons remarqué que les utilisateurs baissaient les bras dès que possible, preuve que la possibilité de baisser les bras après avoir bloqué le mouvement était bien nécessaire pour éviter une fatigue trop rapide. Les utilisateurs moins habitués à la navigation 3D ont moins

profité de cette fonction et donc ont montré des signes de fatigues plus rapidement. Ce qui nous amène à penser qu'une utilisation prolongée serait à terme fatigante.

Q2 - Besoin du retour video (l'utilisateur passe du retour vidéo des contrôles à l'écran de navigation).

R2 - Tout les utilisateurs sans exceptions se sont servis du retour des commandes régulièrement, ce qui nous permet de penser qu'une intégration (plus ou moins) discrète dans l'écran de navigation serait profitable. De plus certains utilisateurs nous ont fait part de ce besoin.

Q3 - Hésitation sur les mouvements, sur les zones d'interaction (l'utilisateur se reprend souvent).

R3 - Une explication détaillée est indispensable aux utilisateurs novices, tandis que ceux plus habitués à la navigation 3D se sont contentés d'indications simples.

Q4 - Utilisation des deux mains, coordinations (une main statique, inversion des mouvements).

R4 - Sur ce point, une différence s'établit entre les novices et les utilisateurs plus habitués. Les premiers n'utilisent en général qu'une seule main, l'autre restant statique mais parfois présentes à l'écran ce qui peut engendrer des mouvements non désirés.

Les seconds, par contre, ont rapidement utilisé leur deux mains simultanément.

Q5 - Sensibilité des mouvements (trop grands, trop petits) et mouvements involontaires (point de vue tremblant?).

R5 - Du fait de la possibilité de paramétrer la sensibilité de chaque commande, les problèmes de mouvements trop rapides ou trop lents ne se sont pas posés.

Par ailleurs, la contiguïté de la zone d'accélération et de celle de rotation a engendré beaucoup de mouvements involontaires.

Q6 - Manière de fermer les mains (pas fermées complètement, une diode reste encore visible, notamment le pouce).

R6 - Nous avons remarqué qu'il y avait un problème avec le pouce comme nous le présentions. En effet, les utilisateurs ne referment pas totalement leurs mains, ce qui laisse la diode apparente. Un autre cas présentant le même problème est le cas où l'utilisateur serre le poing, mais qu'il ne le conserve pas face à la caméra (tranchant de la main perpendiculaire au plan de la caméra). De plus, la fixation précaire des diodes sur le gant leur laissait une certaine liberté qui les laissaient parfois visibles même le gant fermé.

Q7 - Naturel des mouvements bloqués (l'utilisateur oublie de redescendre la main pour arreter d'avancer par exemple).

R7 - Encore une fois, les utilisateurs expérimentés s'adaptent beaucoup plus vite au levier de vitesse que les novices. Il est à noter que le repositionnement du levier au centre oblige l'utilisateur à anticiper l'arrêt de ce type de mouvement.

Q8 - Mains face à la caméra pour le point de vue (l'utilisateur tourne le bras autour de lui).

R8 - L'utilisateur ne translate pas sa main, mais la fait tourner naturellement autour de son épaule. Ceci provoque une disparition des diodes et donc la perte de leur position par le système de reconnaissance. Pour ce qui est de l'autre main, cela peut provoquer un déplacement latéral non désiré.

Q9 - Nombres de doigts utilisés (avant et après avoir signalé que toute la main n'est pas nécessaire).

R9 - Au début de leur expérimentation, les utilisateurs ont utilisé naturellement les cinq doigts, mais au fur et à mesure ils ont préféré utiliser à un seul doigt. Mais cela leur a demandé un petit effort de concentration supplémentaire.

Partie V : Conclusion

En partant d'un sujet libre, mettre en valeur la salle d'immersion 3D du CSTB, nous avons mené un travail d'étude et de réalisation complet. Malgré quelques défauts mineurs au niveau du prototype tels que: une navigation peu intuitive au départ pour des novices de ce domaine, une légère instabilité de la capture provoquant des tremblements du point de vue, le prototype a satisfait tout le monde dans l'ensemble. On peut donc en déduire que les études réalisées antérieurement à la réalisation du prototype, nous ont permis d'éviter les écueils principaux et de concevoir un prototype répondant aux principales attentes des utilisateurs. Même si certaines parties de ces études, en particulier l'UAN, étaient peu adaptées à ce type de projet, d'autres telles que la recherche de l'existant pour la communication gestuelle nous ont permis de définir un ensemble de mouvements relativement intuitif.

Nous tenons à remercier Diane LINGRAND, Anne-Marie DERY-PINNA, Alain GIBION pour leur participation aux tests et leur encadrement, mais également tout l'équipe de la salle d'immersion du CSTB pour leur accueil et la bonne ambiance lors des tests.

Références :

1. Spécification et génération de gestes naturels: application à la langue des signes française - Thierry Lebourque - Université Paris XI – 1998.
2. Vision par ordinateur pour l'interaction homme-machine fortement couplée - François Bérard - UNIVERSITE JOSEPH-FOURIER-GRENOBLE I – 1999.
3. Projet BreakBit : jeu de danse par reconnaissance video de mouvements - Patrick Benoit, Nicolas Joly, François Poyer, Christophe Soum, Baptiste Valliere - Projet ESSI 2 - Juin 2003.
6. Jesse Driver, Rob Read, Eric Blough, Kenneth Seah. An Evaluation of the Polhemus, Spaceball, and Mouse Devices for 3D Cursor Positioning. Computer Science Department, University of Texas at Austin, August, 1990.
10. Ken Hinckley, Randy Pausch, John C. Goble, Neal F. Kassel. A Survey of Design Issues in Spatial Input, Actes de UIST, 1994, pp 213-222. Available as Technical Report TR-90-29.
13. Paul Kabbash, William Buxton et Abigail Sellen. Two-Handed Input in a Compound Task, Actes de CHI'94, pp 417-423.
20. B. Schneiderman. Direct manipulation : A step beyond programming languages. IEEE Computer Numéro 16(8), pp 57-69, 1983.
23. Dan Venolia. Facile 3D Direct Manipulation. Actes de InterCHI'93, pp 31-36, 1993.
25. Colin Ware, Danny R. Jessome. Using the Bat : A Six-Dimensional Mouse for Object Placemen, IEEE Computer Graphics and Applications, 8(6), nov 1988, pp 65-70.
26. Tetsuya Yoshimura, Yasuhiro Nakamura, Masataka Sugiura. 3D Direct Manipulation Interface : Developement of the Zashiki-Warashi System, Computer and Graphics, 18(2), pp 201-207, 1994.
28. Shumin Zhai. User Performance in Relation to 3D Input Device Design, ACM Computer Graphics, 32(4), novembre 1998, pp 50-54.