

Inspection dynamique de modèles CAO

J. Dugalais E. Piccuezzu

seemage

2229 route des Crêtes, 06560 VALBONNE – France

{jd, ep}@seemage.com

Résumé

Nous présentons dans ce qui suit un dispositif d'interface homme machine innovant permettant à l'utilisateur de visualiser intuitivement et dynamiquement le contenu et l'agencement des modèles CAO de grande taille.

Nous présentons ensuite les résultats de l'implémentation de ce dispositif dans un logiciel d'animation 3D temps réel de modèles CAO existants qui démontrent la pertinence des concepts à la base du dispositif.

Mots clefs

Visualisation, Visibilité, Opacité, OpenGL®

1 Introduction

Le domaine d'application du dispositif exposé ci-après est celui de la visualisation et des processus interactifs 3D.

Alors que l'ensemble des logiciels de visualisation 3D existants proposent une gestion explicite de la visibilité et de l'opacité des objets, soit pièce par pièce, soit à travers la gestion de groupes ou de calques, le dispositif présenté propose une gestion implicite de ces attributs et permet ainsi à l'utilisateur de littéralement plonger au cœur de ses modèles 3D à l'aide d'un simple "cliquer-déplacer".

Éliminant ainsi de nombreuses opérations de sélection et de modifications d'attributs, ce dispositif assure un gain de productivité très important à l'utilisateur lors de la manipulation de ses modèles CAO.

2 Présentation du dispositif

2.1 Etat de l'art

On distinguera par la suite l'opacité de la visibilité. L'opacité définit un degré de transparence de l'objet, la visibilité est un état booléen (visible / invisible). L'opacité et la visibilité sont des attributs complémentaires de l'objet.

De nombreuses solutions de visualisation 3D tirent aujourd'hui parti d'une 4ème composante Alpha adjointe

aux 3 canaux classiques de la couleur rouge, vert et bleu. Cette composante Alpha est supportée par les API 3D standards tel OpenGL®.

Dans ces applications, l'attribut d'opacité doit le plus souvent être définie par l'utilisateur. Celui-ci désigne un ou plusieurs objets et leur affecte une valeur de transparence. Il en est de même pour la visibilité, souvent gérée à l'aide de calques ("layers") dans lesquels l'utilisateur navigue pour isoler tel ou tel type d'objet à l'affichage.

Nous faisons le constat suivant :

- Il est très laborieux d'affecter judicieusement des transparences pertinentes sur un grand nombre d'objets
- Il n'existe pas d'outils rapides et intuitifs permettant de modifier rapidement des attributs de visibilité, pour accéder aux pièces internes et cachées au sein de modélisations complexes, et d'accroître ainsi la productivité
- l'opacité et la visibilité ne sont pas dépendantes de la position de l'observateur
- lorsque de nombreux objets sont transparents, l'interprétation du modèle devient très difficile
- il n'est pas possible de limiter ces attributs d'opacité et de visibilité à certaines portions de l'écran

2.2 Description

Le dispositif que nous mettons en œuvre affecte des transparences et modifie la visibilité des objets selon une fonction de la distance $f(\text{objet, caméra})$, et ce dans une ou plusieurs zones 2D limitées de l'espace écran. Ces attributs (opacité / visibilité) calculés automatiquement ne viennent en aucun cas se substituer définitivement aux attributs définis par l'utilisateur.

L'utilisateur choisit sa fonction f (linéaire, log, sin, etc.) et en modifie les paramètres à l'aide de contrôles graphiques intuitifs ("sliders"). La/les vues sont mise à jour en temps réel à chaque modification de la fonction ou d'un de ses paramètres.

Un des intérêts de ce dispositif est qu'il autorise un réglage fin et à la volée de l'opacité et de la visibilité sur un nombre illimité d'objets, en fonction du point de vue.

En jouant sur le type de la fonction f ainsi que sur ses paramètres, l'utilisateur visualise les objets à la profondeur souhaitée. Nous qualifions de profondeur la distance minimale d'un objet à l'écran.

Le contexte du modèle est toujours conservé, puisque l'outil se limite à une ou plusieurs zones de l'écran qui peuvent être déplacées à loisir. Cette zone d'influence peut être fixée pour travailler par la suite avec les outils CAO classiques. Par exemple, les objets complètement transparents ou invisibles dans la zone d'influence ne peuvent pas être sélectionnés.

Le dispositif permet également de "verrouiller" les attributs nouvellement définis afin de les conserver lors des prochaines modifications de la caméra. Le dispositif est aussi pourvu d'un mode "poursuite" dans lequel la zone d'influence 2D suit les objets verrouillés lorsque le positionnement de la caméra est modifié.

2.3 Intégration dans le processus d'affichage

Le dispositif se superpose à l'affichage et surcharge les événements utilisateurs (événements clavier/souris) afin d'assurer sa gestion propre : sélection de la fonction, modification de la profondeur à l'aide du curseur, déplacement de la zone et modification de l'affichage si nécessaire.

Lorsque le dispositif n'a pas le focus, tous les processus interactifs dans l'aire client sont conservés : highlighting, picking, sélection, édition/création d'objets, mouvements de caméras etc.

Dès que le curseur de la souris s'approche d'un élément d'IHM du dispositif (bouton de sélection de la fonction/disque/poignée de modification du rayon/curseur pour la modification du seuil), l'élément se met en surbrillance ("highlighting"), et l'outil prend le focus dans l'aire client de la fenêtre OpenGL®

Dans la zone d'influence, seul les objets visibles peuvent être sélectionnés.

Un rendu spécifique à la zone d'influence peut être réalisé, comme par exemple une illumination propre au dispositif.

2.4 Mise en œuvre

Il s'agit d'un rendu multi passes présenté en Figure 1 et utilisant le "stencil buffer", aujourd'hui supporté par de très nombreux matériels graphiques et API 3D.

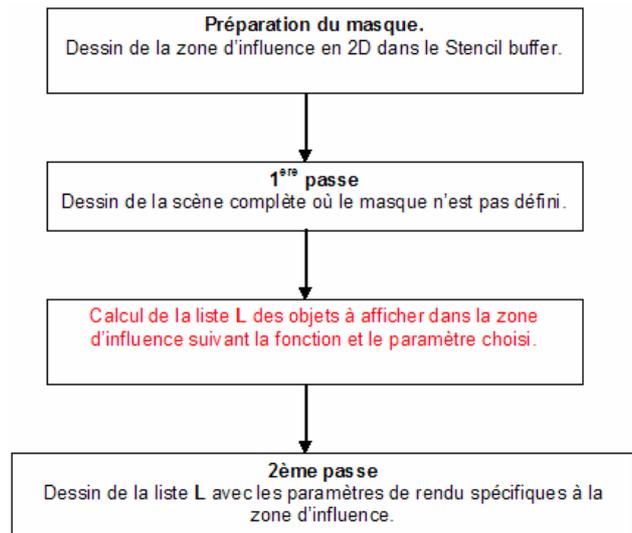


Figure 1 – Rendu multi passes

Ce dispositif est accéléré matériellement sur la plupart des cartes graphiques OpenGL® du marché, même d'entrée de gamme, ce qui lui confère d'excellentes performances.

Les optimisations possibles sont nombreuses. Les performances du traitement rendent cet outil fonctionnel sur tout système de visualisation à base de polygones.

Bien que décrit en s'appuyant sur l'API 3D OpenGL, le dispositif peut être porté sur n'importe quel système et API pourvus d'un "stencil buffer" ou de tout autre mécanisme aux fonctions analogues (pochoir).

Le concept peut également être porté et adapté à la visualisation 3D à base de voxels (volumique).

Le principal intérêt de ce mécanisme est qu'il se substitue en partie seulement aux mécanismes de visualisation existants sur des portions limitées de l'espace 2D. Tous les processus interactifs existants sont conservés. L'outil vient ainsi enrichir grandement les processus de visualisation et de sélection des pièces internes sur des modélisations complexes, sans perturber l'ergonomie originale.

2.5 Optimisations

De nombreuses optimisations visant à limiter le nombre d'objets dessinés peuvent être appliquées pour accélérer la synthèse de l'image. [1, 2]

Les objets peuvent être pourvus de boîtes englobantes dont on détermine si oui ou non elles se projettent dans les zones masquées. Lorsque tous les points de la boîte sont projetés dans la zone masquée, il est inutile de dessiner l'objet qui n'apparaîtra pas.

Notons que d'autres optimisations plus sophistiquées peuvent faire appel à des décompositions spatiales de type octree [3] permettant d'extraire rapidement les objets se projetant dans des zones 2D de l'écran (intersection de l'octree et du cône infini couvrant le disque).

3 Implémentation : seemage Digger®

Afin de valider les concepts présentés auparavant, nous avons implémenté ce dispositif au sein de notre logiciel d'animation 3D temps réel seemage Presenter®.

Conçu pour être accessible à toute personne non spécialiste de la visualisation 3D, seemage Presenter® permet la scénarisation temps réel de présentations multimédias et de contenus e-Training contenant une forte composante d'animation de modèles CAO 3D existants.

Le résultat de l'implémentation des concepts présentés auparavant se nommé seemage Digger® et se présente sous la forme d'un cercle délimitant la zone d'influence et la zone externe, comme représenté sur la Figure 2.

La Figure 1 illustre les éléments de l'interface homme-machine de seemage Digger®:

- Le cercle permet de modifier la position de seemage Digger® dans la fenêtre OpenGL®,
- Le bouton Poignée permet de redimensionner le diamètre de la zone d'influence,
- Le bouton Lampe permet d'activer une illumination propre à Digger qui modifie le rendu dans la zone d'influence dans perturber les éclairages pré existants dans la scène,
- Le bouton Curseur permet de modifier le seuil de la fonction f du dispositif,
- Les boutons Onion Skin, Scanner, Cutting Plane et Zoom permettent d'activer les modes correspondants.

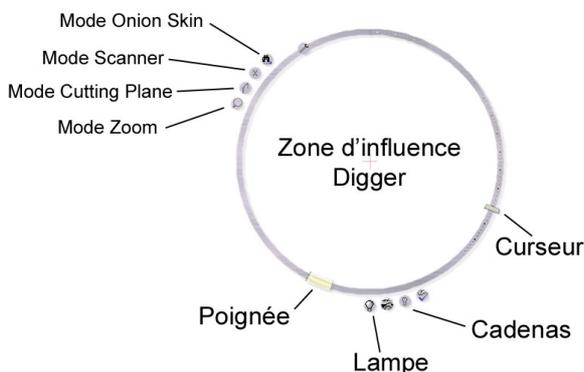


Figure 2 – IHM seemage Digger®

3.1 Mode "Scanner"

La Figure 3 illustre le mode "Scanner" de seemage Digger® qui permet de plonger dans le modèle sans avoir à masquer tout ou partie du modèle.

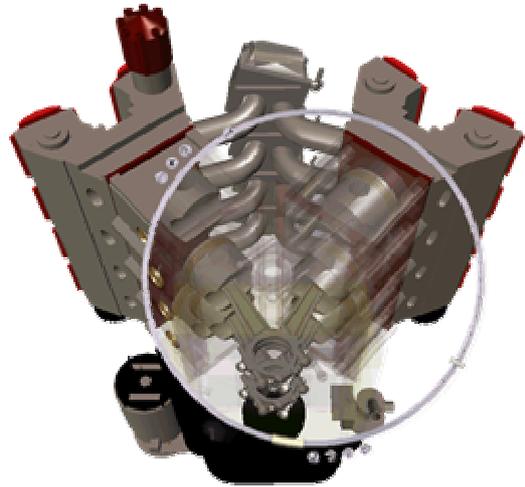


Figure 3 – seemage Digger® : mode SCANNER

Dans ce mode, les objets se voient attribuer un attribut d'opacité en fonction de leur profondeur et d'une valeur fixée par le curseur.

3.2 Mode "Onion Skin"

Le mode "Onion Skin" de seemage Digger® permet de "peler" le modèle couche par couche en ajustant la position du curseur, accédant ainsi instantanément aux objets internes, comme présenté sur la Figure 4:

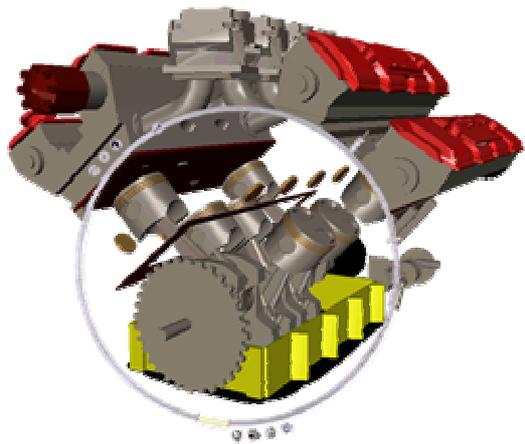


Figure 4 – seemage Digger® : mode ONION SKIN

Dans ce mode, seuls les objets dont la profondeur est supérieure au seuil déterminé par le curseur sont visualisés.

3.3 Mode "Cutting Plane"

La figure 5 illustre une séquence type d'utilisation du mode "Cutting Plane" de seamage Digger®. Ce mode permet de définir et de manipuler un plan de coupe dynamiquement à l'aide du curseur :

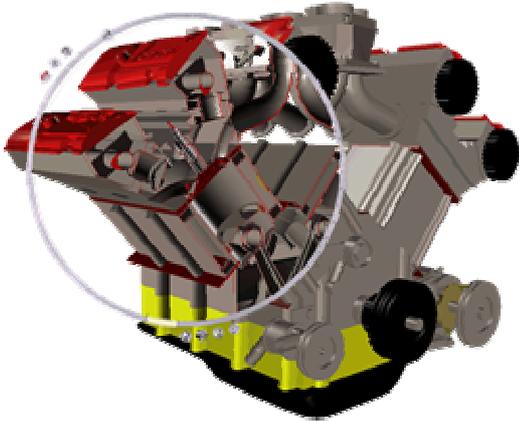


Figure 5 – seamage Digger® : mode CUTTING PLANE

L'utilisateur définit un plan de coupe parallèle au plan de visualisation au moment de l'activation de ce mode, et en modifie la position dynamiquement à l'aide du curseur.

3.4 Mode "Zoom"

Le mode "Zoom" permet à l'utilisateur de disposer d'une loupe centrée automatiquement sur la position du curseur souris au moment de l'activation de ce mode.

Le grossissement de la loupe est réglé dynamiquement par l'utilisateur à l'aide du curseur Digger®, comme représenté en Figure 6:

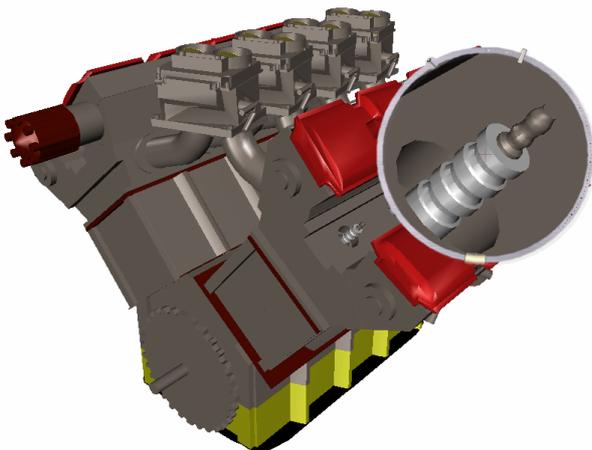


Figure 6 – seamage Digger® : mode ZOOM

4 Conclusion

Nous avons présentés dans ce document un dispositif d'interface homme machine innovant et un exemple d'implémentation concret dans un logiciel d'animation 3D temps réel.

Un des intérêts de ce dispositif est de pouvoir s'intégrer sans difficultés majeures au sein de toute application de visualisation 3D polygonale (API 3D OpenGL, Direct 3D ou autres)

Les champs d'application sont très vastes :

- Mode de visualisation intégrable à toute plate-forme de visualisation CAO 3D basé OpenGL (la quasi-totalité) ou autre, augmentation de la productivité sur des modélisations complexes,
- exploration et découverte de modèles 3D complexes,
- présentation, vues de détail, e-training,
- « radiographie » de modèles 3D,
- imagerie 3D médicale (en complément du contouring),
- etc.

Le dispositif présenté dans ce document fait l'objet d'un dépôt de brevet auprès de l'INPI. sous le numéro 02/13118.

Références

- [1] Dirk Bartz. Large Model Visualization. Techniques and Applications. PhD Thesis, 2001
- [2] Dirk Bartz, Michael Meißner, Tobias Hüttner. OpenGL-assisted Occlusion Culling of Large Polygonal Models. Computers and Graphics, Vol 23, No. 5, 1999
- [3] J. Revelles, C. Urena, M. Lastra. An Efficient Parametric Algorithm for Octree Traversal. Dpt. Lenguajes y Sistemas Informaticos, E.T.S. Ingenieria Informatica University of Granada, Spain