

Exercice 1 : Etude de méthodes de simplification avec MeshLab

Il ya une limite à la complexité des objets que vous pourrez utiliser dans vos scènes 3D. Certains outils comme ZBrush peuvent rapidement créer un nombre de face élevé. La réduction de polygones permet de réduire ces modèles après coup, la meilleure méthode étant de demander à l'infographiste de le faire dès le départ. L'outil Open source MeshLab offre une excellente alternative pour le faire. MeshLab est disponible pour Windows, OSX et Linux.

L'image ci-dessous montre les résultats d'un test qui avait 480 000 faces. Jusqu'à 120 000 faces, la différence de qualité est à peine perceptible. Ci-dessous, vous verrez le modèle devenir de plus en plus rugueux.

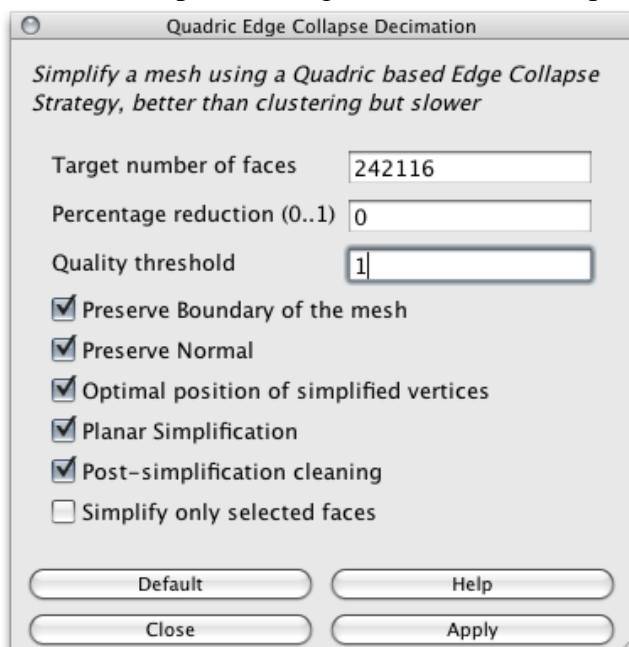


Une première utilisation (in english) (extrait de https://www.shapeways.com/tutorials/polygon_reduction_with_meshlab)

If you haven't already done so, please download and install MeshLab and import your model.

From the menu, select Filters > Remeshing, simplification and construction > Quadratic Edge Collapse Detection. If your model is textured, there is also an option (with texture) that will do a good job at keeping your textures positioned properly. A panel with a few options will show up.

Here are the option settings we believe to be optimal:



Quality threshold: 1. Enter a value between 0 and 1 here; the higher the value the harder MeshLab tries to stick to your original model's shape. The documentation isn't clear on what the consequence of using a higher value is - my impression is that it's slightly slower than low values, so I happily used a value of 1 with great results.

- ➔ Quality threshold affects the simplification penalizing bad shaped faces. To approximate accurately the original shape only with well-shaped triangles you require a higher number of faces with respect to allowing more freedom in the final triangle shape. The value is in the range [0..1]: 0 accept any kind of face (no penalties), 0.5 penalize faces with quality < 0.5, proportionally to their shape.

Preserve Boundary of the Mesh: Yes.

- ➔ The simplification process tries not to destroy mesh boundaries, e.g. exposed edges of the mesh are left untouched. This parameter has no effect on watertight meshes.

Preserve Normal: Yes.

- ➔ Select this to stop MeshLab from accidentally flipping the face normals. Try to avoid face flipping effects and try to preserve the original orientation of the surface. The only drawback of enabling it is a slight increase in the processing times. On by default.

If you DO run into inverted normals issues when uploading, try reorienting the normals with the option Filter > Normals, Curvature and Orientation > Re-Orient all faces coherently (note that this will only work for manifold objects).

Optimal position of simplified vertices: Yes.

- ➔ When collapsing an edge the chosen vertex position minimizes the current estimated error. If disabled, the edges are collapsed onto one of the two vertices and the vertices of the final mesh are a subset of the original mesh. It's on by default.

Planar simplification: Yes.

- ➔ Add additional simplification constraints that try to preserve the current shape of the triangles. It can greatly improve the quality of the shape of the final triangles on perfectly planar portions of the mesh. Like the quality threshold it affects the accuracy/complexity ratio. Off by default because it is very useful only in particular situations like when you have perfectly flat areas finely tessellated."

- 1- Utiliser cette méthode sur deux types d'objets
 - a. Objets n'ayant pas d'arêtes vives (exemple Bouddha)
 - b. Objet ayant des arêtes vives (par exemple une maison ou un immeuble)
- 2- Comparer les méthodes utilisables dans MeshLab (en prenant en compte préférentiellement celles présentées dans le cours). Cette comparaison doit se faire dans un fichier word comprenant vos commentaires et des captures d'écran
- 3- Trouver des modèles qui posent des problèmes pour ces méthodes (arêtes vives, changement de topologie, ...).

Exercice 2 : Simplification par partitionnement

Ecrire la méthode de simplification par cellule vu en cours :

- construire une grille,
- fusionner les sommets,
- modifier les triangles.

Tester la méthode sur les maillages du TP précédent.

Faire varier la taille de la grille pour faire varier la résolution des maillages.

Remarque : des éléments pouvant vous aider dans la compréhension se trouvent dans le répertoire tp09 .zip et le fichier tp09.pdf

Exercice 3 (pour en savoir plus) : Utilisation de Qslim

Objectif : Illustrer le cours par l'utilisation d'un programme de simplification de triangulation basé sur la méthode QEM

A faire individuellement ou par binôme, sous Windows.

Compte-rendu sous forme d'un fichier PDF (par exemple créé à l'aide de Open Office). Il faut bien sûr expliquer ce que vous faites (texte + captures d'écran).

Récupérer l'archive (TP09_bis.zip) contenant les logiciels et fichiers de données nécessaire pour ce TP puis la décompacter dans votre répertoire de travail.

Démarrer l'invite de commandes DOS et se placer dans le répertoire de travail qui doit contenir les fichiers suivants :

- des fichiers triangulation au format SMF
- des fichiers image au format PGM raw (image en niveaux de gris)
- les programmes exécutables :
 - o QSlim.exe : le programme de simplification de triangulation de Michael Garland basé sur la méthode QEM
 - o vrmlview.exe : un programme de visualisation 3D utilisant le format VRML (site correspondant)
 - o pgm2smf.exe : conversion du format PGM au format SMF (fichier source)
 - o smf2vrml.exe : conversion du format SMF au format VRML (fichier source)
 - o la librairie cygwin1.dll nécessaire pour les exécutables pgm2smf.exe et smf2vrml.exe .

Plus d'informations sur la méthode peuvent être trouvées sur le site :

<http://mgarland.org/research/thesis.html>

1) Un fichier de données triangulation est au format SMF, fichier texte avec :

- en début de fichier le mot `begin`
- la liste des sommets, pour chaque sommet la lettre `v` suivie des 3 coordonnées
- la liste des triangles, pour chaque triangle la lettre `f` suivie des indices des 3 sommets (indice à partir de 1)
- en fin de fichier le mot `end`

Les différents fichiers fournis sont :

- `mannequin.smf` (1355 triangles)
- `hypersheet.smf` (897 triangles)
- `bunny.smf` (69451 triangles)
- `Horse.smf` (39698 triangles)
- `plane.smf` (228317 triangles)
- `dragon.smf` (200000 triangles)
- `happy.smf` (200000 triangles)

Pour visualiser un fichier SMF avec VRMLView, il faut d'abord le convertir avec le programme QSlim, par exemple en tapant la commande suivante :

```
QSlim -j -o mannequin.vrml -M vrml mannequin.smf
```

puis pour le visualiser

```
vrmlview mannequin.vrml
```

VRMLview permet de visualiser un seul fichier (modèle) à la fois.

Les rotations, translations et zooms se font à la souris avec les boutons resp. gauche, milieu et droite.

Le menu File permet entre autres :

- de charger un nouveau fichier (`Open model ...`)
- de rafraichir le modèle actuel (`Refresh model`) si le fichier correspond a été modifié,
- de revenir à un modèle déjà chargé, (`Models, Next, Previous`)
- de faire une copie d'écran (`Save snapshot ...`)

Le menu View permet de modifier des paramètres de visualisation avec notamment :

- la visualisation (ou non) des faces (`Filled`)
- la visualisation (ou non) des arêtes (`Wireframe`)

L'item `Line Color` du menu `Settings` permet de changer la couleur des arêtes.

A titre d'exercice, entraînez-vous à visualiser les différents modèles correspondants aux fichiers SMF fournis.

2) Le programme `QSlim` permet de simplifier un fichier SMF en fixant le nombre de triangles désiré. Par exemple, pour simplifier le modèle `Horse.smf` de 39698 à 10000 triangles, exécuter la commande suivante :

```
QSlim -t 10000 -o res.vrml -M vrml Horse.smf
```

puis pour le visualiser

```
vrmlview res.vrml
```

Pour avoir les différentes options de `QSlim`, taper la commande :

```
QSlim -h
```

Par exemple, dans la commande de l'exemple de simplification ci-dessus :

```
-t 10000      pour fixer le nombre final de triangles
-o res.vrml   pour indiquer le nom du fichier résultat
-M vrml       pour avoir le fichier résultat directement au format VRML
```

L'option `-j` permet de préserver le modèle initial sans faire de simplification, par exemple pour convertir un fichier SMF en fichier VRML (cf. exemple de la partie 1)

Testez la simplification sur les différents modèles. Pour un même modèle, comparer les résultats obtenus en choisissant deux simplifications différentes.

3) Par défaut le programme `QSlim` utilise la suppression d'arêtes (edge collapse) en plaçant les nouveaux sommets de manière optimale et en préservant si possible les bords de la triangulation (arêtes externes)

Les options `-B` et `-O` permettent de modifier le comportement du programme :

- en utilisant le modèle `hypersheet.smf`, comparer le résultat obtenu en utilisant ou non l'option `-B 0`
 - par défaut préservation des bords :
`QSlim -t 200 -o res1.vrml -M vrml hypersheet.smf`

- utilisation de l'option -B 0 qui indique de traiter les arêtes externes comme les arêtes internes
 QSlim -B 0 -t 200 -o res2.vrml -M vrml hypersheet.smf
- en utilisant le modèle mannequin.smf, comparer le résultat obtenu en utilisant ou non l'option -O 0
 - par défaut placement optimal des nouveaux sommets :
 QSlim -t 400 -o res1.vrml -M vrml mannequin.smf
 - utilisation de l'option -O 0 : lors de chaque suppression d'arête, la position du nouveau sommet correspond à un sommet de l'arête supprimée
 QSlim -O 0 -t 400 -o res2.vrml -M vrml mannequin.smf