Qualité de surface et lissage

Noura Faraj *noura.faraj@umontpellier.fr*

Objectif

Dans cet exercice, vous implémenterez les éléments suivants :
• Calcul de l'opérateur laplacien uniforme et son utilisation pour le lissage (lissage uniforme de Laplace)
• Calcul de l'opérateur de Laplace-Beltrami et utilisation de celui-ci pour l'approximation de la courbure moyenne et pour le lissage (lissage dans la direction normale)
• Calcul du rapport de longueur du bord circonférence au minimum pour évaluer les formes de triangle d'un maillage. Comparer l'effet des deux algorithmes de lissage en termes d'influence sur la forme du triangle.
• Approximation de la courbure gaussienne.

Décompresser le fichier, dans le dossier résultant :
make -j
./tp

Question 1

a) L'opérateur uniforme de Laplace se rapproche du Laplacien de la surface discrétisée en utilisant le centroïde du un voisinage. Pour un sommet v les n sommets voisins sont noté v_i. L'approximation uniforme laplacienne est :

$$L_U(v)=\left(\frac{1}{n}\sum_i v_i\right)-v$$

La moitié de la longueur du vecteur $L_U(v)$ donne une approximation de la courbure moyenne.

Implémentez une fonction calc_uniform_mean_curvature (). Cette fonction doit remplir une propriété vunicurvature_ du vertex avec la courbure moyenne approximation en utilisant l'opérateur uniforme de Laplace.

b) Implémentation du lissage uniforme de Laplace dans une fonction uniform_smooth (unsigned int _iters). Il faut appliquer des opérations de lissage _iters sur le maillage, où une opération de lissage déplace chaque sommet du maillage à mi-chemin le long de son vecteur $L_U(v)$:

$$v'=v+\frac{1}{2}L_U(v)$$

Astuce : Assurez-vous que tous les sommets sont lissés en parallèle. N'oubliez pas de mettre à jour les normales de vertex après le changement de coordonnées de sommet.

c) Ajouter la fonction :

```
void addNoise(){
   for( unsigned int i = 0 ; i < vertices.size() ; i ++ )
   {
      float factor = 0.03;
```

```
      const Vec3 & p = vertices[i];
      const Vec3 & n = normals[i];
      vertices[i] = Vec3( p[0] + factor*((double)(rand()) / (double)
(RAND_MAX))*n[0], p[1] + factor*((double)(rand()) / (double)
(RAND_MAX))*n[1], p[2] + factor*((double)(rand()) / (double)
(RAND_MAX))*n[2]);
   }

}
```

et ajouter le bruit au chargement du maillage :

```
openOFF("data/elephant_n.off", mesh.vertices, mesh.normals, mesh.triangles,
mesh.triangle_normals);
mesh.computeNormals();
mesh.addNoise();
```

Testez l'effet du lissage en faisant plusieurs itération. Qu'observez vous ? Commenter par rapport au bruit ajouté sur le maillage d'entrée.

d) Vous aurez remarqué que les maillages perdent du volume lorsqu'on les lisse avec le filtre implémenté. Un filtre très similaire permet de limiter cet effet : il s'agit du filtre de Taubin. Ce dernier prend deux paramètres : λ et μ, où $0 < \lambda < -\mu$. Une itération du lissage se divise en deux :
- une itération du filtre précédent avec λ paramètre
- une itération du filtre précédent avec μ pour paramètre
Implémentez ce filtre dans une méthode taubinSmooth (float lambda , float mu) .
Testez le sur différents exemples.
Qu'observez vous ?

Notes :
— L'auteur préconise des valeurs de L'auteur préconnise des valeurs de λ et μ proches en valeur absolues : 0.330 et -0.331 par exemple.
— Pour de plus amples informations, voir Curve and surface smoothing without shrinkage, Gabriel Taubin, ICCV 1995.

Question 2

De nombreuses applications nécessitent des maillages triangulaires avec des triangles bien formés. Triangles équilatéraux sont généralement considérés comme de bonne qualité, les triangles maigres ou plats sont « mauvais ». Une mesure de cette qualité est le rapport entre le rayon du cercle circonscrit de triangle et la plus petite longueur d'arête. Plus ce rapport est petit, plus le plus le triangle est proche du triangle équilatéral (idéal). Pour dériver une formule pour le rayon du cercle circonscrit, on peut utiliser ces deux expressions pour l'aire d'un triangle :

$$A = \frac{|a|.|b|.|c|}{4\,r} = \frac{|a \times b|}{2},$$

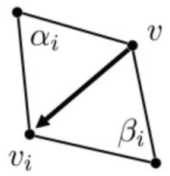
Ou a, b et c sont les vecteurs représentant les arêtes des triangles, tels que a et b partagent un sommet à comme origine.

Implémentez une fonction calc_triangle_quality () de sorte qu'elle remplisse une propriété de face propriété tshape_ avec ce rapport pour chaque triangle. (Astuce : pour la stabilité numérique, assurez-vous que si votre dénominateur croisé est petit ou négatif, vous affectez simplement une valeur élevée à la forme en triangle mesure et ne la calcule pas avec la formule générale.)

Pour les maillages irréguliers, le lissage uniforme de Laplace déplace les sommets non seulement suivant la normale à la surface, mais aussi tangentiellement. Pour créer un opérateur de lissage qui déplace les sommets suivant la normale, on peut utiliser l'opérateur Laplace-Beltrami. Cet opérateur utilise très des poids très spécifiques pour les sommets voisins :

$$L_B(v) = \sum_i w_i(v_i - v) \quad = \sum_i (\cot \alpha_i + \cot\beta_i)(v_i - v)$$

Où les 2 angles sont ceux opposés à l'arête entre v et v_i :



Le vecteur de Laplace-Beltrami donne une approximation de la normal et sa demi-longueur une approximation de la courbure moyenne à ce sommet.

a) Ecrire une fonction calc_weights() calculant les poids cotangents.
Implémentez l'approximation de courbure moyenne en utilisant l'opérateur de Laplace-Beltrami. Ecrire une fonction calc_mean_curvature () remplissant une propriété vcurvature_ avec la courbure moyenne approximé.

b) Effectuez le lissage à l'aide de l'opérateur Laplace-Beltrami. Normaliser les poids cotangents des arêtes pour que leur somme soit égale à la somme l'unité :

$$L_B = \frac{1}{\sum_i w_i} \sum_i w_i (v_i - v)$$

N'oubliez pas d'utiliser le facteur d'amortissement de $\frac{1}{2}$comme dans le cas d'un lissage uniforme. Comparez les effets des deux lissages en visualisant la forme des triangles.

Question 3

Lors du cours, nous avons vu un moyen facile d'approcher la courbure gaussienne pour un sommet d'un maillage triangulaire. C'est simplement le défaut d'angle - en utilisant la somme des angles autour d'un sommet :

$$G = 2\pi - \sum_j \theta_j$$

Implémentez une fonction calc_gauss_curvature () qu'elle stocke les approximations de courbure gaussienne dans une propriété de sommet vgausscurvature_.

Bonus

Mettre en œuvre un « lissage tangentiel » qui déplace les sommets uniquement dans le plan tangent du sommet, concentre donc (uniquement) sur l'amélioration des formes en triangle. Pour cela, vous devez projeter l'uniforme Laplace revient en moyenne au plan tangent du sommet. Notez que vous devez calculer et stocker la normale initiale du sommet, afin de garder les sommets toujours actifs le plan tangent d'origine, même après plusieurs itérations de lissage tangentielles.