Laplacian Curvature Enhance

Alexander Pinzon*
Cimalab Research Group

Eduardo Romero†
Cimalab Research Group

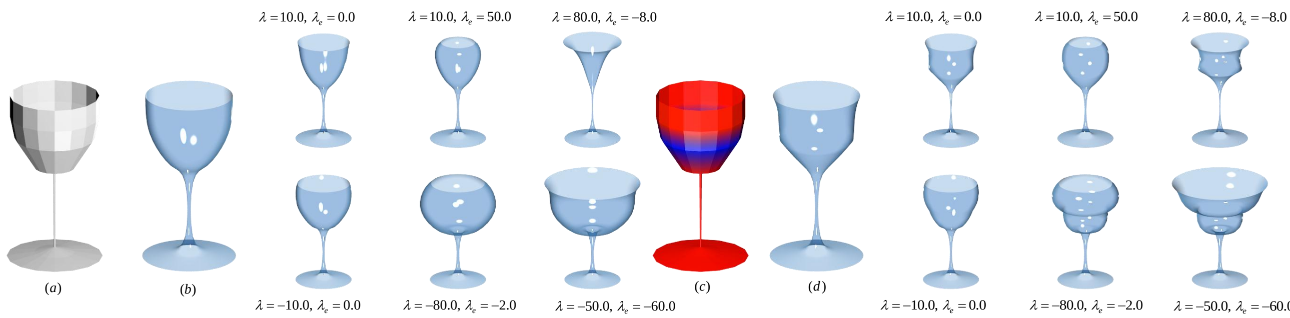

Figure 1: *Family of cups generated with our method from coarse model to enhancing the curvature obtained from Catmull-clark Subdivision and the use of constraints over coarse model with weigth vertex group in red.*

Abstract

This paper proposes a novel method for modelling polygonal mesh using curvature enhancing. This method use our extension of Laplace Beltrami operator for triangles and quads meshes to enhance global curvature in the model. This work present a novel applications of curvature enhancement in sculpting and modelling with subdivision surfaces and weight vertex groups. We show a series of graphics examples that demonstrate the quality, predictability and flexibility of our results in a real production environment with software blender.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling —Modeling packages

Keywords: laplacian smooth, subdivision surface

1 Introduction

Over the last years have been developed novel techniques of modeling that can generate a variety of shapes to look natural and realistic [Botsch et al. 2006]. Editing techniques have evolved from affine transformations to advanced tools like scuplting [Coquillart 1990; Galyean and Hughes 1991; Stanculescu et al. 2011], editing and creation from sketches [Igarashi et al. 1999; Gonen and Akleman 2012], complex interpolation techniques [Sorkine et al. 2004; Zhou et al. 2005], among others.

Traditional methods for smooth surfaces from coarse geometry like Catmull-Clark have been widely developed [Catmull and Clark 1978; Stam 1998], these works generalize uniform B-cubic splines knot insertion to meshes, some of them add control of the results with the use of creases to produce sharp edges [DeRose et al. 1998], or the modification of weights on the vertices that control locally the zone of influence [Biermann et al. 2000], instead our method performs a feature enhancement of the model allowing parameterize the curvature of the surface creating a family of different versions of the same object preserving detail and realistic natural look of the original model.

*e-mail: apinzonf@gmail.com
†e-mail:edromero@unal.edu.co

Many types of brushes have been developed to sculpt meshes, brushes that perform inflation lose detail when inflating the vertices [Stanculescu et al. 2011], our method allows inflation of the mesh vertices moving in the opposite direction to the curvature preserving the shape and sharp features of the model.

This work is organized as follows: In the section 2 show works related to the laplacian mesh processing, digital sculpting, and offsetting methods for polygonal meshes; In the section 3 we describe the theoretical framework of the Laplacian operator for polygon meshes; In the section 4 we present our proposed method for curvature enhancement and applications on subdivision surfaces and sculpting; Lastly some results are shown in graphic examples of Laplacian operator for polygonal meshes composed of triangles and quads. Also shows some results of curvature enhancement applications in sculpting, subdivision and modelling methods.

Contributions Our work present an extension of the Laplace Beltrami operator for meshes of arbitrary topology composed for triangles and quads representing a larger spectrum of mesh that works with today eliminating the need for preprocessing and allows the preservation of the original topology. With this operator we proposed a method to generate family of parameterized shapes, in a robust and predictable way. Our method enable to customization of smoothness and curvature obtained during subdivision surfaces process. We proposed a new brush for enhance silhouette features of mesh in modeling and sculpting.

2 Related work

Many tools have been developed for modeling based on Laplacian mesh processing. Thanks to the kindness of the Laplacian operator these tools have in common the need for preservation of the geometric details of the surface for the different processes such as: freeform deformation, fusion, morphing and other aplications [Sorkine et al. 2004].

Methods for offset polygon meshes based on the curvature defined by the Laplace Beltrami operator have been developed. These methods allow adjusting shape offset by a constant distance with high enough precision to minimize Hausdorf error. The problem with these methods is the loss of detail caused by smoothing, which depends on the size of the offset [Zhuo and Rossignac 2012]. In volumetric approaches on computing the offset boundary that are based on distance field computation in point-based representation, this methods the topology of the offset model can be different from that the original geometry [Chen and Wang 2011].

[Gal et al. 2009] proposes automatic features detection and shape edition with feature inter-relationship preservation. In analysis step they define salient surface features how ridges and valleys with base on first and second order curvature derivates [Ohtake et al. 2004], and angle-based threshold. In feature characterization step the curves are classified by several properties as planar or non-planar, approximated by line, circle or ellipse shapes, and so on. In edit step the user define initial change over several feature and then this edit is propagated over other features with base in your inter-relationships. This method works fine with objects that have sharp edges composed of basic geometric shapes such as lines, circles or ellipses but this method has difficulties when models are smoother with organic forms and cannot find the features to edit and preserve.

Digital sculpting is divided into two principal methods: based on polygonal methods and voxel grids-based methods. Brushes for inflate operations in polygonal methods only depends on the normal at each vertex [Stanculescu et al. 2011], in grids-based some operations permit add or remove voxels and then have that processing isosurfaces from volume to produce polygonal meshes representation [Galyean and Hughes 1991]. The problem whit this type of operations is the difficult to maintain surface details during larger scale deformation.

3 Laplacian Smooth

The Laplacian Smooth techniques allows you to reduce noise on a mesh's surface with minimal changes on its shape. Computer graphics objects which have been reconstructed from real world, contain undesirable noise. A laplacian smoothing removes undesirable noise while still preserves desirable geometry as well as the shape of the original model.

The functional used in many laplacian smoothing approach to constrain energy minimization is based on a total curvature of a surface S.

$$E\left(S\right)=\int_S \kappa_1^2+\kappa_2^2 dS \tag{1}$$

Where κ_1 and κ_2 are the two principal curvatures of the surface S.

3.1 Gradient of Voronoi Area

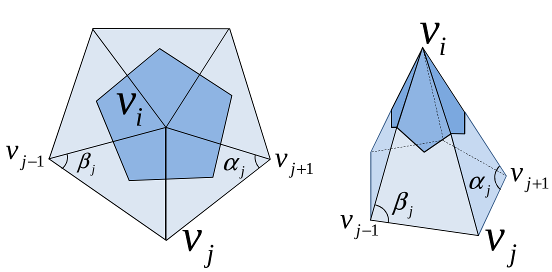

Figure 3: *Area of Voronoi region around v_i in dark blue. v_j 1-ring neighbors around v_i. α_j and β_j opposite angles to edge $\overrightarrow{v_j - v_i}$.*

Consider a surface S compound by a set of triangles around vertex v_i. We can define the *Voronoi region* of v_i as show in figure 3, The change in area produced by move v_i is named gradient of *Voronoi region [Pinkall et al. 1993; Desbrun et al. 1999; Meyer et al. 2003].*

$$\nabla A=\frac{1}{2}\sum_j\left(\cot\alpha_j+\cot\beta_j\right)\left(v_i-v_j\right) \tag{2}$$

If we normalize this gradient in equation (2) by the total area in 1-ring around v_i, we have the *discrete mean curvature normal* of a surface S as shown in equation (3).

$$2\kappa\mathbf{n}=\frac{\nabla A}{A} \tag{3}$$

3.2 Laplace Beltrami Operator

The *Laplace Beltrami operator* LBO denoted $\triangle_g$ is used for measures mean curvature normal of the Surface S [Pinkall et al. 1993].

$$\triangle_g S=2\kappa\mathbf{n} \tag{4}$$

The LBO has desirable features, one feature of the LBO is in direction of surface area minimization, allowing us to minimize energy using it on a total curvature of a surface S at equation (1).

4 Proposed Method

Our method allow the editing of geometric features using the curvature enhancement and smoothing. Generating a parameterized family of shapes using a set of vertices representing a coarse sketch of the desired model. Our approach can be mixed with traditional or uniform subdivision surfaces methods and is iterative and converges towards a continuous and smooth version of the original model.

Unlike other methods, our method allows to use mixed arbitrary types of mesh representation as triangles and quads, exploiting the basic geometrical relationships facilitating and ensuring convergence of the algorithm and similar shapes consistent with the original shape against the other methods.

Our method allows the use of soft constraints weighting the effect of smoothing at each vertex based on a normalized weight, the weights are assigned to the control vertices of the original mesh or. The weights of the new vertices resulting from the subdivisions are calculated by interpolation, allowing to modify the behavior of the method on exact regions of the original model.

Our approach contain an extension of the Laplace Beltrami operator for meshes composed by triangles and quads. Using meshes composed by triangles and quads has been increasing in recent years due to the flexibility of modeling tools as Blender 3D [Blender-Foundation 2012]. Today many artists manually connecting vertices such that its edition allows simplest way to perform animation processes and interpolation [Mullen 2007]. For these reasons it is very important to develop an operator that allows working with this type of mesh immediately, eliminating the need to preprocess the mesh to convert to triangles and losing the original design made by users.

4.1 Laplace Beltrami operator over triangular and quadrilateral meshes TQLBO

Given a mesh $M=(V,Q,T)$, with vertices V, quads Q, triangles T.

The area of 1-ring neighborhood (N_1) with shared face to vertex v_i in M is.

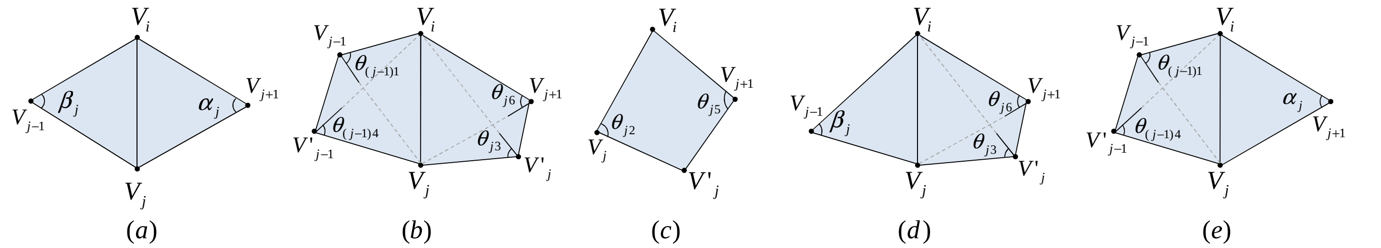

Figure 2: *The 5 basic triangle-quad cases with common vertex V_i and the relationship with V_j and V'_j. (a) Two triangles [Desbrun 1999]. (b) (c) Two quads and one quad [Xiong 2011]. (d) (e) Triangles and quads (TQLBO).*

$$A\left(v_i\right) = A\left(Q_{N_1\left(v_i\right)}\right) + A\left(T_{N_1\left(v_i\right)}\right).$$

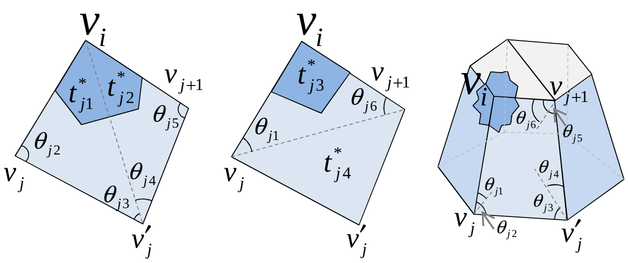

Figure 4: $t^*_{j1} \equiv \triangle\ v_i v_j v'_j$, $t^*_{j2} \equiv \triangle\ v_i v'_j v_{j+1}$, $t^*_{j3} \equiv \triangle\ v_i v_j v_{j+1}$ *Triangulations of the quad with common vertex v_i proposed by [Xiong 2011] to define Mean LBO.*

Appling the mean average area according to [Xiong et al. 2011] of all posible triangulations for each quad to $A\left(Q_{N_1\left(v_i\right)}\right)$ as show in figure 4.

$$A\left(v_i\right) = \tfrac{1}{2^m}\sum_{j=1}^{m} 2^{m-1} A\left(q_j\right) + \sum_{k=1}^{r} A\left(t_k\right)$$

Where $q_1, q_2, ..., q_i, ..., q_m \in Q_{N_1\left(v_i\right)}$ and $t_1, t_2, ..., t_k, ..., t_r \in T_{N_1\left(v_i\right)}$.

$$A\left(v_i\right) = \frac{1}{2}\sum_{j=1}^{m}\left[A\left(t^*_{j1}\right) + A\left(t^*_{j2}\right) + A\left(t^*_{j3}\right)\right] + \sum_{k=1}^{r} A\left(t_k\right) \quad (5)$$

Applying the gradient operator to (5).

$$\nabla A\left(v_i\right) = \frac{1}{2}\sum_{j=1}^{m}\left[\nabla A\left(t^*_{j1}\right) + \nabla A\left(t^*_{j2}\right) + \nabla A\left(t^*_{j3}\right)\right] + \sum_{k=1}^{r} \nabla A\left(t_k\right) \quad (6)$$

According to (2), we have.

$$\nabla A\left(t^*_{j1}\right) = \frac{\cot\theta_{j3}\left(v_j - v_i\right) + \cot\theta_{j2}\left(v'_j - v_i\right)}{2}$$

$$\nabla A\left(t^*_{j2}\right) = \frac{\cot\theta_{j5}\left(v'_j - v_i\right) + \cot\theta_{j4}\left(v_{j+1} - v_i\right)}{2}$$

$$\nabla A\left(t^*_{j3}\right) = \frac{\cot\theta_{j6}\left(v_j - v_i\right) + \cot\theta_{j1}\left(v_{j+1} - v_i\right)}{2}$$

$$\nabla A\left(t_k\right) = \frac{\cot\alpha_k\left(v_k - v_i\right) + \cot\beta_{k+1}\left(v_{k+1} - v_i\right)}{2}$$

All triangles and quads configurations of the 1-neighborhood faces adjacent to v_i can be simplified in five simple cases how show in figure 2.

Then according to equation (3), (4), and five simples cases defined in figure 2 the TQLBO (Triangle-Quad LBO) of v_i is.

$$\triangle_g\left(v_i\right) = 2\kappa\mathbf{n} = \frac{\nabla A}{A} = \frac{1}{2A}\sum_{v_j \in N_1\left(v_i\right)} w_{ij}\left(v_j - v_i\right) \quad (7)$$

$$w_{ij} = \begin{cases} \left(\cot\alpha_j + \cot\beta_j\right) & \text{case } a. \\ \frac{1}{2}\left(\cot\theta_{(j-1)1} + \cot\theta_{(j-1)4} + \cot\theta_{j3} + \cot\theta_{j6}\right) & \text{case } b. \\ \left(\cot\theta_{j2} + \cot\theta_{j5}\right) & \text{case } c. \\ \frac{1}{2}\left(\cot\theta_{j3} + \cot\theta_{j6}\right) + \cot\beta_j & \text{case } d. \\ \frac{1}{2}\left(\cot\theta_{(j-1)1} + \cot\theta_{(j-1)4}\right) + \cot\alpha_j & \text{case } e. \end{cases} \quad (8)$$

We define a Laplacian operator as a matrix equation

$$L\left(i,j\right) = \begin{cases} -\frac{1}{2A_i} w_{ij} & \text{if } j \in N\left(v_i\right) \\ \frac{1}{2A_i}\sum\limits_{j \in N\left(v_i\right)} w_{ij} & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \quad (9)$$

Where L is the $n \times n$ matrix, n is the number of vertices of a given mesh M, w_{ij} is the TQLBO defined in equation (8), $N\left(v_i\right)$ is the 1-ring neighbors with shared face to v_i, A_i is the ring area around v_i.

Normalized version of the TQLBO as a matrix equation

$$L\left(i,j\right) = \begin{cases} -\frac{w_{ij}}{\sum\limits_{j \in N\left(v_i\right)} w_{ij}} & \text{if } j \in N\left(v_i\right) \\ \delta_{ij} & \text{otherwise} \end{cases} \quad (10)$$

Where δ_{ij} being the Kronecker delta function.

4.2 Curvature Enhancing

The curvature enhancing use the change produced by laplacian smoothing in the inverse direction of the curvature flow for moves the vertices in the portions of the mesh with most curvature. In this proccess we use a diffusion proccess:

$$\frac{\partial V}{\partial t} = \lambda L\left(V\right)$$

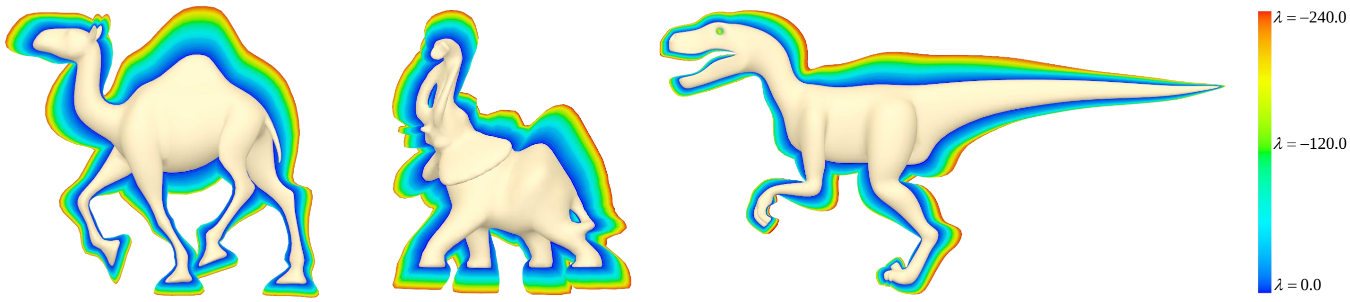

Figure 5: *A set of 48 successive curvature enhance shapes, from* $\lambda = 0.0$ *in blue to* $\lambda = -240.0$ *in red, with steps of* -5.0.

For solve the equation above we use implicit integration and a normalized version of TQLBO matrix.

$$(I - |\lambda dt| \, W_p L) \, V' = V^t \quad (11)$$

$$V^{t+1} = V^t + \text{sign}\,(\lambda) \left(V' - V^t\right)$$

The vertices $V^{t+1}$are enhance along their inverse curvature normal directions by solving this simple linear system: $Ax = b$, where $A = I - |\lambda dt| \, W_p L$, L is the Normalized TQLBO defined in the equation (10), $x = V'$ are the smoothing vertices, $b = V^t$ are the actual vertices positions, W_p is a diagonal matrix with weigth vertex group, sign (x) is the sign function, and λdt is the enhance factor that support negative and positive values, negative for enhancing positive for smoothing.

At the borders of the meshes that are not closed, you can not calculate the curvature, for that reason we use the scale-dependent operator proposed by [Desbrun et al. 1999].

Our method was designed for use with weigth vertex groups to specify the degree of impact on the solution, the weights vary between 0 and 1 with a value of 0 makes no changes and with values 1 applies the total change. The weights modifying influence zones where the Laplacian is applied as shown in the equation 11. The weights on each vertex will produce a different solution for that reason are put before obtaining the solution of the linear system. Families of shapes that are generated may change substantially with the weights of specific control points.

The model volume increases as the lambda is most negative, this can be countered by a simple method of preserving volume. In [Desbrun et al. 1999] present a simple method to resize the mesh but have a problem the model suffer large displacements with $\lambda < -1.0$ or perform multiple iterations. We propose the following solution: If we have v_i^{t+1} is a mesh vertex of V^{t+1} in the $t+1$ iteration.

$$\overline{v} = \tfrac{1}{n} \sum_{v_i \in V} v_i,$$

$\overline{v}$ is the center of the mesh, vol_{ini} is an initial volume, and vol_{t+1} is the volume at the iteration $t+1$, then we have that scale factor for resize de volume is

$$\beta = \left(\tfrac{vol_{ini}}{vol_{t+1}} \right)^{\frac{1}{3}}$$

and the new vertices positios are:

$$v_{i\,new}^{t+1} = \beta \left(v_i^{t+1} - \overline{v} \right) + \overline{v}$$

4.3 Sculpting

We design a new brush that allows enhancement of the curvatures of a polygon mesh in real time. Our brush work well with the stroke method *Drag Dot* developed in the sculpt mode in Blender [Blender-Foundation 2012] which allows you to preview the change that occurs in the model until you release the mouse button or tablet, also enable moves the mouse over the model to fit exactly where you want to perform the enhancement of the curvature.

Brushes that perform similar work as inflate can create distortions in the mesh and can also produce self-intersections of the mesh, as this brushes only moves the vertices along the normal and does not take into account the global information. Whereas our method look for the best way to make inflation while keeping the global curvature for preserving the shape and sharp features of the model.

Our method simplifies the work required for the enhancement, which would be to use some different brushes for inflating and some other to soften and styling. With our enhanced brush can be performed in one step.

For the real-time work brush is necessary that the matrix is constructed with the vertices that are within the radius defined by the user involvement, which dramatically reduces the size of the matrix to be processed, the center of this sphere depends on the place where the user clicks on the canvas and three-dimensional mesh place where the click is projected.

Furthermore special handling is necessary with the vertices are at the boundary of which have neighbors that are not within the radius of impact, these vertices must be marked as boundary and not the curvature calculated for them, but must be present in the matrix to allow the vertices that have all their neighbors within the selection radius to calculate correctly the curvature, this change allows the results to be much smoother on the border. The laplacian matrix for sculpt mode as a matrix equation.

$$L\,(i,j) = \begin{cases} -\frac{w_{ij}}{\sum\limits_{j \in N(v_i)} w_{ij}} & \text{if } \|v_i - u\| < r \wedge \|v_j - u\| < r \\ 0 & \text{if } \|v_i - u\| < r \wedge \|v_j - u\| \geq r \\ \delta_{ij} & \text{otherwise} \end{cases}$$

Where $v_j \in N\,(v_i)$, u is the center of sphere the radius r. Then the matrix should remove rows and columns of vertices index that are not within the radius.

4.4 Subdivision surfaces

The Catmull-Clark subdivision transformation is used to smooth a surface as the limit of sequence of subdivision steps[Stam 1998]. This method do a recursive subdivision transformation that refines the model into a linear interpolation that is a approximate smooth

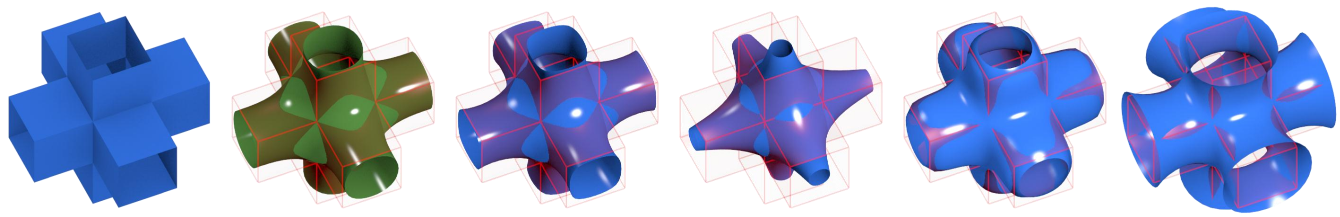

Figure 6: *Left: Original Model, in green color model with Catmull-Clark Subdivision. Models with laplacian smoothing:* $\lambda = 60.0$, $\lambda_e = 12.0$ *and* $\lambda = 90.0$, $\lambda_e = 80.0$. *Models first filter with laplacian smoothing* $\lambda = 60.0$, $\lambda_e = 12.0$ *and before applied curvature enhancing:* $\lambda = -200.0$, $\lambda_e = 12.0$ *and* $\lambda = -40.0$, $\lambda_e = -100.0$.

surface. The process of Catmull-Clark is govern by properties of B-spline curve from multivariate spline theory[Loop 1987]. In many subdivision surfaces methods Catmull-Clark, Loop so on. The smoothness of the model is autmaically guarantteed[DeRose et al. 1998].

Catmull-Clark subdivision surface method generates smooth and continuous models from a coarse model and produce results quickly due to the simplicity of implementation, but with these methods is not easy to make changes to the global curvature of the model. If we use Catmull-Clark subdivision surfaces and curvature enhancement for modeling from sketches with few vertices can generate families of shapes with just a parameter, which would allow an artist to choose the model of a similar set of options that best meets your needs without having to change each one of the control vertices.

Our method allows the use of vertex weigth paint over the control points. The weights can be applied to the coarse model, then to perform subdivision weights are interpolated, producing weights with smooth changes in the zones of influence, so the curvature that is obtained is much softer at those areas. Using vertex group weigth and subdivision surfaces is shown in figure 1.c.

In the equation 11, W_p is a diagonal matrix with weights corresponding to each vertex. The weights at each vertex produce a different solution for this reason the matrix is placed in the diffusion equation, as families that are generated may change substantially with weighted of specific control points.

5 Results

In this section we describe the results of our curvature enhance method that used our extension of Laplace Beltrami operator for triangles and quads TQLBO with several example models in figures 5,8. We test the curvature enhance with TQLBO method on a PC with AMD Quad-Core Processor @ 2.40 GHz and 8 GB RAM.

Figure 7 show the results when applying Laplacian smoothing operator TQLBO of equation (9) in a model with a simple subdivision. In column (c) Laplacian smoothing was applied to the model consists of only quads. In column (d) the model was converted to triangles and then Laplacian smoothing was applied. In column (e) the model is converted randomly some quads into triangles and then Laplacian smoothing was applied, showing similar results to those composed only of triangles or squares.

Figure 8 show the generation of diferent version of camel with the variation of parameter lambda. In the top row you can see results of do curvature enhance over all model, as the lambda becomes more negative curvature in this model makes close in the concave parts and inflates on the convex parts as shown in figure 5. We use negative values to enhance model silhouette features, among more negative the lambda, the model will be further enhanced the silhouette

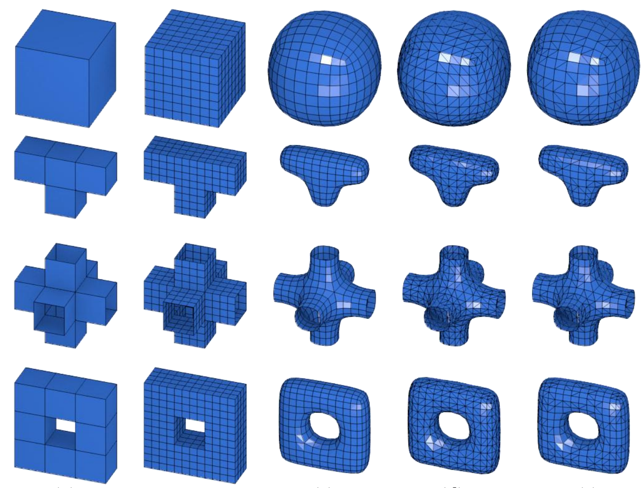

Figure 7: *(a) Original Model. (b) Simple subdivision. (c), (d) (e) Laplacian smoothing with* $\lambda = 7$ *and 2 iterations: (c) for triangles, (d) for quads, (e) for triangles and quads random chosen.*

features. In the bottom row of the figure 8 can be observed using weighted vertex groups, which allows you to specify which areas you want to model enhancement, on the left is the enhancement of the legs of the camel produces enhancement of organic aspect, also notes that the border is not distorted and smooth in the union.

Nosotros realizamos pruebas del operador laplaciano con la ecuacion (9) y su version normalizada ecuacion (10), las dos producen resultados similares si los triangulos que componen la malla son del mismo tama;o en promedio, pero la version normalizada es mucho mas estable y predecible, debido a que no es dividida por el area del anillo que puede producir problemas de calculo debido a erroreres de punto flotante como se observa en la figura 9 (c) bottom row en la cual la malla se deformo pues el TQLBO es suceptible al tamaño de los triangulo. El realce de curvatura con el laplaciano normalizado tiene un comportamiento mas regular. El modelo se puede deformar en la version normalizada del TQLBO con lambdas grandes > 400 se autointersecta, pero no produce los picos que se observan con el TQLBO. En la figura 9 se observan resultads diferentes debido a que el area de los triangulos en este modelo no es regular de manera que donde hay triangulos de mayor area el realce es menor (figura 9 (c) skull), y donde los triangulos son mas pequenios se produce un realce mayor (figura 9 (c) chin).

Nuestro metodo para realizar el realce de las silouete features es predecible e invariante frente a transformaciones isometricas como las presentes en animaciones, en esta animacion se muestran algunas poses del camello realizando una caminata. En esta animacion las patas y el cuello son las partes que se les realiza en realce como se observa en la parte izquierda abajo de la figura11. Modificaciones locales producidas con metodos como pose interpolation or rigging animation no afectan significativamente el resultado como en el caso de las patas del camello cada pose muestra una flexion diferente de las articulaciones de las patas del camello el realce permite mantener flesh-like shapes producidas en el modelo original por el artista. Esto se debe al proceso de difusion al cual es sometida la malla de forma que peque;os cambios locales son tratados globalmente sin que afecte significativamente la solucion. Nuestro metodo es invariante de rotacion pues unicamente depende del normal field of the mesh, wish is invariant under global rotations.

En la figura 10 vemos el uso del curvature enhancing de forma local en tiempo real. Se uso una sola pasada de la brocha tipo stroke dot de blender como se muestra en la figura con el radio azul y rojo. En la figura 10 (b) se observa como en la parte de arriba la pata del camello se autointersecta y se observa como dos burbujas pegadas lo mismo sucede con los dedos de la mano en la parte de abajo. Con el uso del realce de silouete features en la figura 10 (c) vemos resultados mejores pues no se perdio la forma de la silueta y se conservaron los detalles de los dedos y la pata. Resultados similares pueden ser oobtenidos por un artista pero le tomaria varios pasos y el uso de varias brocas con el realce de curvaturas solo toma un paso. Con esta nueva tecnica se pueden realzar facilmente los musculos en personajes organicos durante el proceso de sculpting.

El uso de metodos de subdivision surfaces como catmull-clark con el realce permite modificar la curvatura que se obtiene con el proceso de subdivision como se observa en la figura (1) en la cual se utilizo a coarse model of cup, despues se le aplico la subdivision de superficies y luego se realizo suavizado y realce laplaciano con la modificacion de los parametros λ y λ_e que corresponden a los lambda para anillos y para bordes respectivamente. En la figura (1) (c), (d) se observa ademas el uso weigth vertex groups sobre coars model luego se le realizo subdivision de superficies con lo cual los pesos de los nuevos vertices eran interpolados con estos nuevos pesos se realizo el realce obteniendo las 6 copas que se encuentran a la derecha de la figura (1) (d). El suavizado laplaciano aplicado con subdivision simple puede producir resultados similares a los obtenidos con Catmu en modelos cuyos triangulos son en promedio iguales como se observa en la figura 6 el modelo en color verde y el obtenido con laplacian smoothing $\lambda = 60.0$, $\lambda_e = 12.0$, pero ademas puede modificar la curvatura obtenida luego de aplicar Catmull-Clark como se muestra en las 3 columnas de la derecha de la imagen 6.

5.1 Implementation

Our method was implemented how a modifier and brush on the blender software [Blender-Foundation 2012] with C and C++, trabajar con blender nos permitio probar de forma interactiva el metodo junto con otros como catmull clark, weigth vertex groups y el sistema de sculpting implemetados en blender .

Para mejorar el rendimiento se trabajo con la mesh struct de bajo nivel de blender calculando todos los datos posible por cada triangle or quad visited y sumandolos a su correspondiente indice en una lista que almacenaba la suma de los laplacianos del anillo, de esta manera solo se debia recorrer dos veces la lista de caras del modelo y dos veces la lista de bordes si la malla no era cerrada. Para la brush fue necesario crear una lista que mapeara los indices de los vertices seleccionados a una lista de 1 a N donde N es el numero de vertices selecccionados y tambien el numero de filas en el sistema lineal a resolver, con esto se reducian drasticamente los calculos que se debian realizar permitiendo trabajar con la herramienta en tiempo

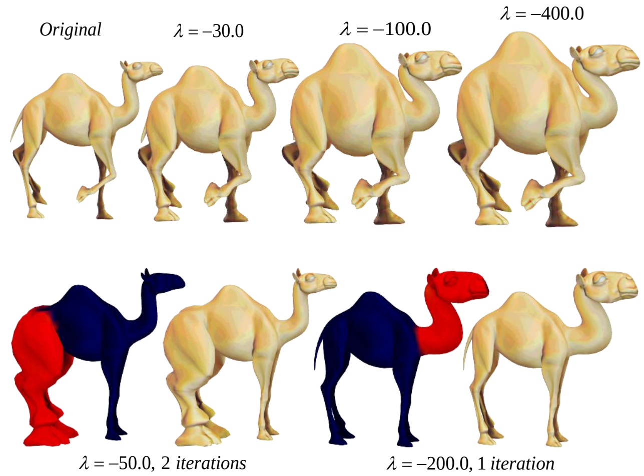

Figure 8: *Top row: Original camel model in left. Curvatture enhancing with $\lambda = -30.0$, $\lambda = -100.0$, $\lambda = -400.0$. Bottom row: Curvature enhancing with weigth vertex group, $\lambda = -50.0$ and 2 iterations at legs, $\lambda = -200.0$ and 1 iteration in head and neck.*

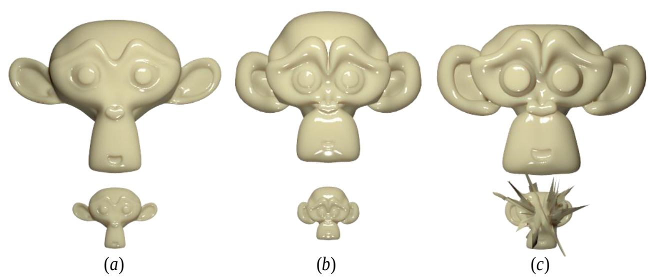

Figure 9: *(a) Top row: Original model scaled by 4. Bottom row: Original Model (b) Top and bottom row: enhancing with Normalized-TQLBO $\lambda = -50$ (c) Top and bottom row: enhancing with TQLBO $\lambda = -50$.*

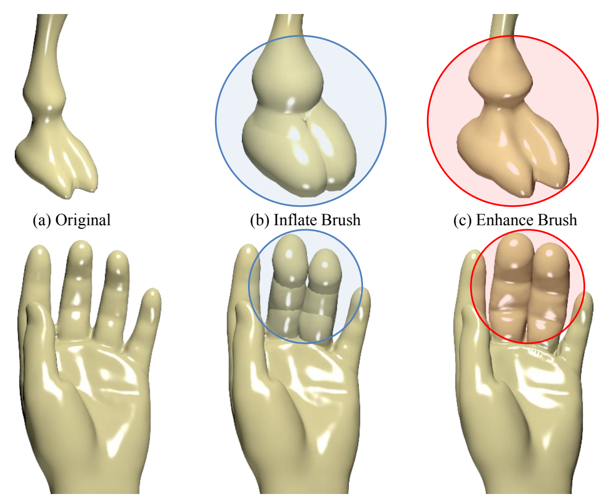

Figure 10: *Top row: (a) Leg Camel, (b) Inflate brush for leg into blue circle, (c) Enhance curvature brush for leg into red circle. Bottom row: (a) Hand, (b) Inflate brush for fingers into blue circle, (c) Enhance curvature brush for fingers in red circle.*

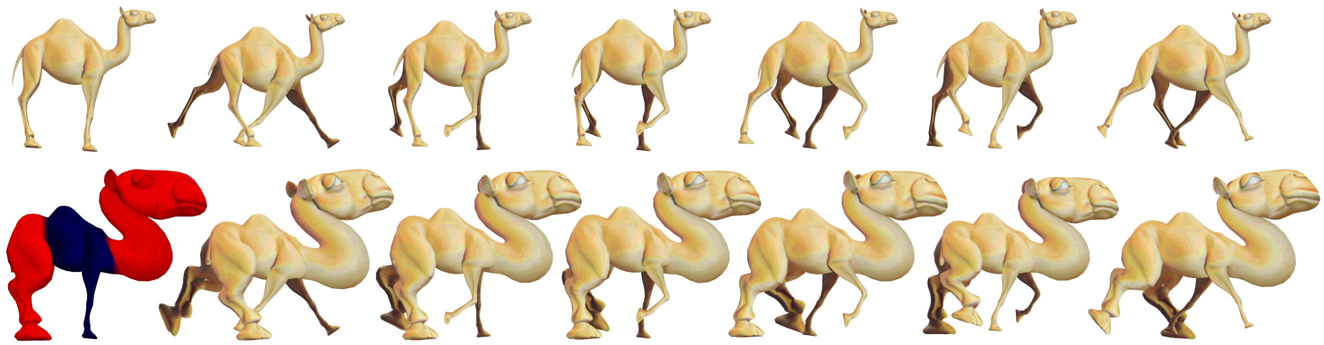

Figure 11: *Our method is pose insensitive. The enhanced for the different poses are similar in terms of shape. Top row: Original walk cycle camel model. Bottom row: Curvature enhancing with weigth vertex group,* $\lambda = -400$ *and* 2 *iterations.*

real. En la construccion de nuestro matri laplaciana se bloquearon lo vertices que tenian caras o bordes vecinos con areas o bordes valor 0 que pueden provocar picos y malos resultados.

La matrix 9 es sparse pues el numero de vecinos por vertice que corresponde al numero de datos por fila es pequeño en comparacion con el numero total de vertices en la malla. Para resolver el sistema lineal de la ecuacion 11 se utilizo OpenNL wich is a a library for solving sparse linear system.

6 Conclusion and future work

Nosotros presentamos una extension del operador laplace beltrami para triangulos y quads que permite trabajar en ambientes de produccion sin necedidad de conversion que ofrece resultados similares a los que se obtendrian al trabajar unicamente con triangulos o con cuadrados. Nosotros contribuimos el concepto de realce de las siluetas en una malla durante el modelamiento o sculpido que permite realizar en pocos pasos la modificacion de la curvatura de un modelo manteniendo su forma general.

Nosotros introducimos una nueva metodo de modelamiento y mostramos algunos de sus posibles usos. Mostramos que el metodo se comporta de forma predecible lo cual facilitara los procesos de aprendizaje, ademas se demostro que el metodo trabaja bien con transformaciones isometricas abriendo la posibilidad de introducirlo en las etapas de animacion.

Nosotros demostramos que esta herramienta sirve para trabjar en las primeras etapas en la que se usan coarse models modificando la curvatura generada por la sub de sup con cc, evitando la edicion de los vertices al tener unicamente que cambiar unos pocos parametros.

Como trabajo futuro nos gustaria analizar teoricamente la relacion entre la subdivision de superficies con catmull clark y el suavizado laplaciano dado que en algunos casos pueden producir resultados muy similares pero la subdivision con catmull es un metodo muy rapido lo que permitiria reducir los tiempos de calculo para obtener la curvatura en un modelo.

Acknowledgements

CIM&LAB Computer Imaging & Medical Applications Laboratory at Universidad Nacional de Colombia.

Blender Foundation.

This work was supported in part by the Google Summer of code program at 2012. Livingstone elephant model is provided courtesy of INRIA and ISTI by the AIM@SHAPE Shape Repository. Hand model is courtesy of the FarField Technology Ltd.

References

BIERMANN, H., LEVIN, A., AND ZORIN, D. 2000. Piecewise smooth subdivision surfaces with normal control. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, SIGGRAPH '00, 113–120.

BLENDER-FOUNDATION, 2012. Blender open source 3d application for modeling, animation, rendering, compositing, video editing and game creation. http://www.blender.org/.

BOTSCH, M., PAULY, M., ROSSL, C., BISCHOFF, S., AND KOBBELT, L. 2006. Geometric modeling based on triangle meshes. In *ACM SIGGRAPH 2006 Courses*, ACM, New York, NY, USA, SIGGRAPH '06.

CATMULL, E., AND CLARK, J. 1978. Recursively generated b-spline surfaces on arbitrary topological meshes. *Computer-Aided Design 10*, 6 (Nov.), 350–355.

CHEN, Y., AND WANG, C. C. L. 2011. Uniform offsetting of polygonal model based on layered depth-normal images. *Comput. Aided Des. 43*, 1 (Jan.), 31–46.

COQUILLART, S. 1990. Extended free-form deformation: a sculpturing tool for 3d geometric modeling. *SIGGRAPH Comput. Graph. 24*, 4 (Sept.), 187–196.

DEROSE, T., KASS, M., AND TRUONG, T. 1998. Subdivision surfaces in character animation. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, ACM, New York, NY, USA, SIGGRAPH '98, 85–94.

DESBRUN, M., MEYER, M., SCHRÖDER, P., AND BARR, A. H. 1999. Implicit fairing of irregular meshes using diffusion and curvature flow. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, ACM Press Addison-Wesley Publishing Co., New York, NY, USA, SIGGRAPH '99, 317–324.

GAL, R., SORKINE, O., MITRA, N. J., AND COHEN-OR, D. 2009. iwires: An analyze-and-edit approach to shape manipulation. *ACM Transactions on Graphics (Siggraph) 28*, 3, #33, 1–10.

GALYEAN, T. A., AND HUGHES, J. F. 1991. Sculpting: an interactive volumetric modeling technique. *SIGGRAPH Comput. Graph. 25*, 4 (July), 267–274.

Gonen, O., and Akleman, E. 2012. Smi 2012: Short paper: Sketch based 3d modeling with curvature classification. *Comput. Graph. 36*, 5 (Aug.), 521–525.

Igarashi, T., Matsuoka, S., and Tanaka, H. 1999. Teddy: a sketching interface for 3d freeform design. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, SIGGRAPH '99, 409–416.

Loop, C. 1987. *Smooth Subdivision Surfaces Based on Triangles*. Department of mathematics, University of Utah, Utah, USA.

Meyer, M., Desbrun, M., Schröder, P., and Barr, A. H. 2003. Discrete differential-geometry operators for triangulated 2-manifolds. In *Visualization and Mathematics III*, H.-C. Hege and K. Polthier, Eds. Springer-Verlag, Heidelberg, 35–57.

Mullen, T. 2007. *Introducing character animation with Blender*. Indianapolis, Ind. Wiley Pub. cop.

Ohtake, Y., Belyaev, A., and Seidel, H.-P. 2004. Ridge-valley lines on meshes via implicit surface fitting. *ACM Trans. Graph. 23*, 3 (Aug.), 609–612.

Pinkall, U., Juni, S. D., and Polthier, K. 1993. Computing discrete minimal surfaces and their conjugates. *Experimental Mathematics 2*, 15–36.

Sorkine, O., Cohen-Or, D., Lipman, Y., Alexa, M., Rössl, C., and Seidel, H.-P. 2004. Laplacian surface editing. In *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, ACM, New York, NY, USA, SGP '04, 175–184.

Stam, J. 1998. Exact evaluation of catmull-clark subdivision surfaces at arbitrary parameter values. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, ACM, New York, NY, USA, SIGGRAPH '98, 395–404.

Stanculescu, L., Chaine, R., and Cani, M.-P. 2011. Freestyle: Sculpting meshes with self-adaptive topology. *Computers & Graphics 35*, 3, 614 – 622. Shape Modeling International (SMI) Conference 2011.

Xiong, Y., Li, G., and Han, G. 2011. Mean laplace-beltrami operator for quadrilateral meshes. In *Transactions on Edutainment V*, Z. Pan, A. Cheok, W. Muller, and X. Yang, Eds., vol. 6530 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 189–201.

Zhou, K., Huang, J., Snyder, J., Liu, X., Bao, H., Guo, B., and Shum, H.-Y. 2005. Large mesh deformation using the volumetric graph laplacian. *ACM Trans. Graph. 24*, 3 (July), 496–503.

Zhuo, W., and Rossignac, J. 2012. Curvature-based offset distance: Implementations and applications. *Computers & Graphics 36*, 5, 445 – 454.