Incremental Catmull-Clark Subdivision

Hamid-Reza Pakdel
University of Calgary
hrpakdel@cpsc.ucalgary.ca

Faramarz Samavati
University of Calgary
samavati@cpsc.ucalgary.ca

Abstract

In this paper, a new adaptive method for Catmull-Clark subdivision is introduced. Adaptive subdivision refines specific areas of a model according to user or application needs. Naive adaptive subdivision algorithm change the connectivity of the mesh, causing geometrical inconsistencies that alter the limit surface. Our method expands the specified region of the mesh such that when it is adaptively subdivided, it produces a smooth surface whose selected area is identical to when the entire mesh is refined. This technique also produces a surface with an increasing level of detail from coarse to fine areas of the surface. We compare our adaptive subdivision with other schemes and present some example applications.

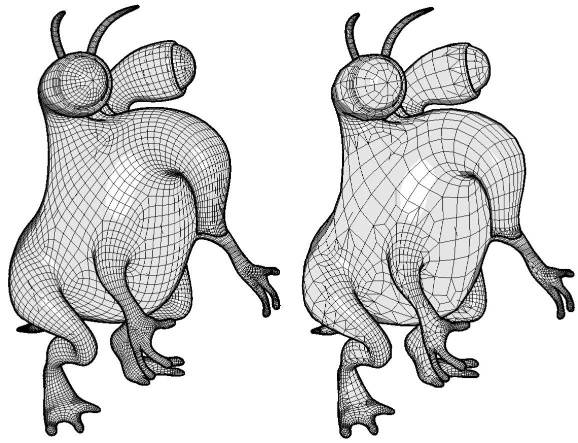

Figure 1. Global and incremental subdivisions

1. Introduction

Subdivision surfaces have become a commonly used tool in modeling and animation packages. Subdivision is defined by simple operations that are globally applied to a given control mesh. Repeated applications of these operations produce a sequence of meshes that converge to a smooth limit surface. Using special subdivision rules, it is possible to create piecewise smooth surfaces. Subdivision surfaces have a clear advantage over NURBS and Bézier tensor product patches, that are traditionally used in computer modeling and animation applications, because subdivision operators can be applied to arbitrary topology two-manifold meshes. In addition, subdivision surfaces do not have the continuity limitations that tensor product patches have when connecting multiple patches to produce piecewise smooth surfaces [5].

In many cases, subdivision of the entire input mesh is not necessary nor desired. Recursive subdivision exponentially increases the number of faces of the mesh, and leads to heavy computational load at higher levels of subdivision. Generally high curvature areas require more subdivisions than planar regions in order to obtain a smooth surface. In modeling applications designers may require a detailed view of a portion of the model that they are working on. Adding features to a mesh requires an increase in the level of detail where the feature is being added. In such cases, adaptive subdivision produces a surface with lesser face count by subdividing only some areas of the mesh.

Naive adaptive subdivision introduces connectivity inconsistencies that must be addressed for correct rendering, editing and further processing of the mesh. Because the shape of the limit surface depends on the connectivity and the geometry of the vertices, care must be taken to avoid any side-effects to the subdivision scheme. Repeated adaptive subdivision may lead to a large difference in the level of subdivision of adjacent faces. It is desirable that the variation in subdivision depth between adjacent faces are limited [12].

Later we will describe how current algorithms deal with the issues that arise when some faces of a mesh are subdivided, including their disadvantages. Our contribution in this paper is a new adaptive subdivision scheme that is designed to address the limitations of these algorithms within the constrains that we have established to obtain both an efficient algorithm and a visually pleasing surface. *Incremental subdivision*, our method, produces meshes with consistent connectivity and geometry. An incrementally subdivided region of a mesh resembles the same area of the surface produced by globally subdividing the mesh. Incremental subdivision limits the subdivision depth of adjacent faces, so the surface gradually varies from coarse to fine. An example of a globally and incrementally subdivided model is shown in Figure 1.

The local nature of subdivision operations naturally lends itself to a data structure that allows efficient and intuitive local modifications to the mesh. Such a data structure allows the subdivision operation to be performed entirely within the graph space of the mesh. One of the goals in designing incremental subdivision was to avoid methods that would require additional book-keeping, including methods that require off-line pre-computation and storage of the subdivision levels.

We introduced incremental Loop subdivision in an earlier paper [9]. Loop subdivision is an approximating scheme that operates on triangular meshes [7]. The incremental Loop subdivision is very efficient as well as easy to implement. However, it is limited to triangular meshes. Our contribution in this paper is the extension of incremental subdivision to arbitrary faces. We have focused on Catmull-Clark subdivision [4], which can be applied to general topology meshes with arbitrary faces. It has a simple face-splitting refinement algorithm, produces a surface that approximates the control mesh, and is $\mathbf{C}^2$ almost everywhere.

In section 2 we give an overview of the Catmull-Clark subdivision. Existing adaptive subdivision schemes are covered in section 3. Incremental subdivision is covered in detail in section 4. Results and applications of incremental subdivision are presented in section 5.

2. Catmull-Clark Subdivision

In Catmull-Clark subdivision, mesh M^i is subdivided to produce the mesh M^{i+1} by splitting each face of the mesh to m quadrilaterals, where m is the number of vertices of the face, and repositioning the vertices of M^i. As illustrated in Figure 2, three kinds of vertices are created for each face split. *Face-vertex* f_j^{i+1} is the centroid of the face that includes vertices v^i, v_j^i, and v_{j+1}^i. *Edge-vertex* e_j^{i+1} is computed as

$$e_j^{i+1} = \frac{1}{4}(v^i + v_j^i + f_j^{i+1} + f_{j-1}^{i+1}), \quad (1)$$

where v_j^i is the jth neighbour of vertex v^i. Subscripts are taken modulo n; the valence of v^i. *Vertex-vertex* v^{i+1} is computed by repositioning vertices of M^i:

$$v^{i+1} = \left(\frac{n-2}{n}\right) v^i + \frac{1}{n^2}\sum_j^{n-1} v_j^i + \frac{1}{n^2}\sum_j^{n-1} f_j^{i+1}. \quad (2)$$

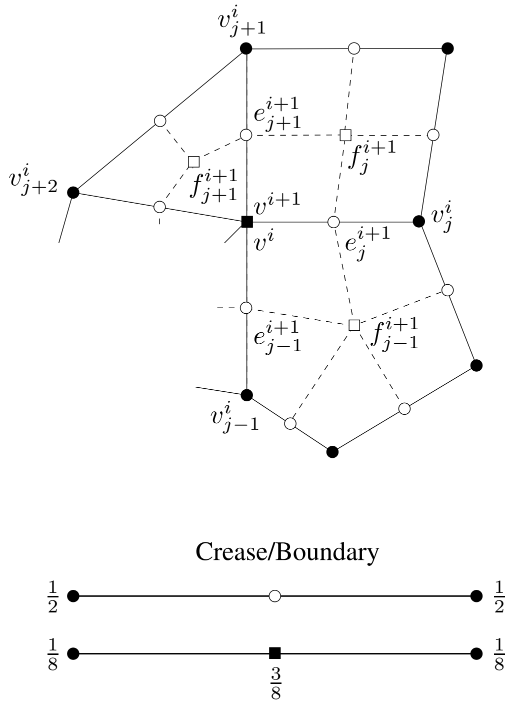

Figure 2. Catmull-Clark subdivision. The • denotes existing vertex, □ denotes face-vertex, ○ denotes edge-vertex, and ■ denotes vertex-vertex.

An example of Catmull-Clark subdivision is given in Figure 3. To produce piecewise smooth surfaces, a connected series of edges are tagged as sharp and subdivided so that the surface is tangent continuous along these edges, but not across them [3, 5]. Boundary edges are crease edges with only one face. Figure 2 also shows the mask of edge-vertices and vertex-vertices along a crease or boundary edge.

3. Adaptive Catmull-Clark Subdivision

3.1. Selecting the Subdivision Area

Adaptive subdivision involves selecting the areas of the mesh require refinement. This decision can be made either by the application or the user. Higher curvature regions of the mesh require more subdivisions than planar areas. Incidentally, higher curvature areas contain more details than flat areas. *Dihedral angle*, the angle between the face normals of adjacent faces, provides a sufficient measure of the surface curvature. In real-time visualization, other factors such as frustum visibility, distance to the viewer, and the pixel area of faces, may be used to tune the selection algorithm. For example, a view frustum visibility test can determine whether a face should be refined [1].

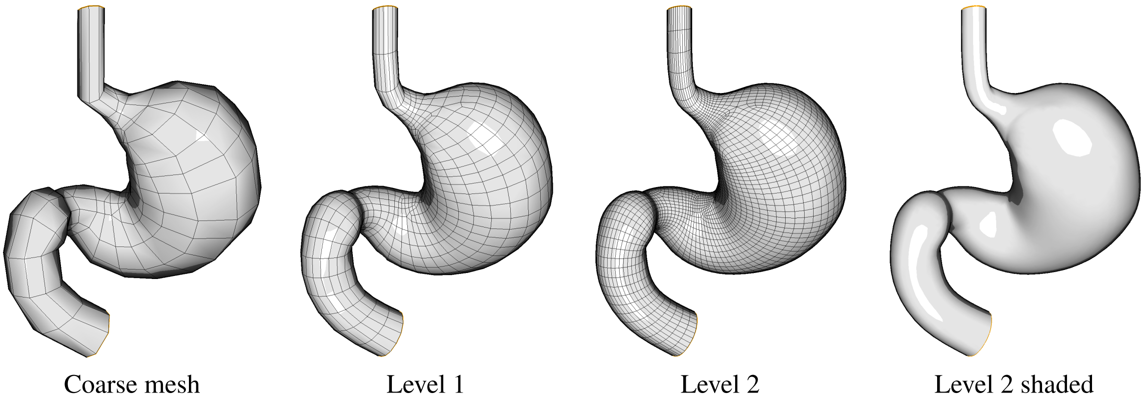

Figure 3. Catmull-Clark subdivision

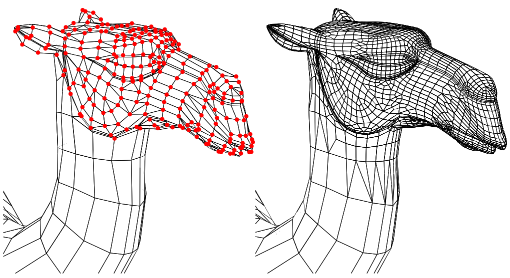

Figure 4. Adaptive subdivision of a user selected area as indicated by the highlighted vertices on the left

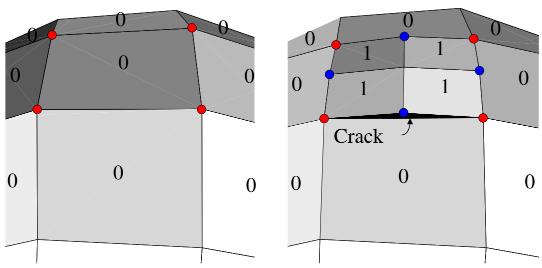

Figure 5. Cracks due to subdivision depth difference of adjacent faces. The numbers represent the current subdivision depth of each face

In rendering applications, the selection algorithm may be derived by non-geometric properties of the mesh. For example, to generate smooth silhouettes the selection criteria can be set to take the normal of each face and the view vector into consideration and subdivide all the faces that share edges on the silhouette boundary [6]. In non-photorealistic rendering methods that are based on edge size [11], the selection area can be determined by the size of the coarse edges.

In modeling applications, users may need local control over the level of detail of the model. Artists sometimes emphasize part of a scene by increasing the detail of that area. Lastly, adding features to a mesh generally necessitates an increase in the level of detail where the features are being added. In these cases, the subdivision area is determined by the user. Figure 4 shows adaptive subdivision of a user defined area.

3.2. Handling Cracks

Though the idea of adaptive subdivision is simple, care must be taken to avoid connectivity inconsistencies that arise as the result of subdividing a subset of faces of the mesh. Figure 5 shows what happens when only one face of the mesh is subdivided. Edge-vertices between faces with different subdivision depths create cracks when they are repositioned in the subdivision process. Cracks must be removed from the mesh for editing, rendering, and processing of the mesh. Since the subdivision algorithm depends on the connectivity of the mesh, it is also crucial that cracks are removed after each subdivision step.

The most common method to remove cracks is to insert new edges that connect the edge-vertices on the cracks to the other vertices within the face with lower subdivision depth [8]. We call these newly introduced edges *crack removing edges* (CREs). Figure 6 shows two possible cases where cracks may appear in a face and how to remove them. When a face has only one crack, it is triangulated by connecting the edge-vertex that caused the crack to the rest of the vertices in the face. If a face has two cracks, then the crack is removed either by triangulating the face or by inserting a face-vertex in the face and connecting it to the vertices to produce quads. This later approach is preferred

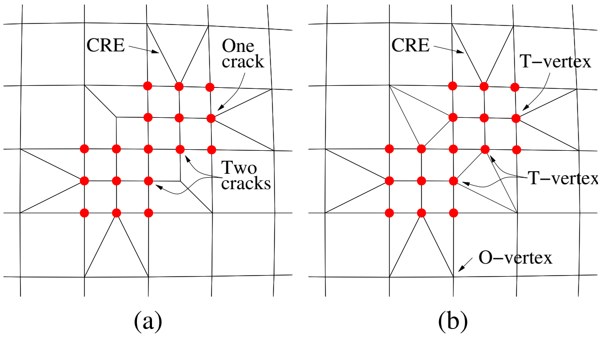

Figure 6. Inserting edges to remove cracks. Two possible methods exist when there are two cracks in a face: (a) quads (b) triangles.

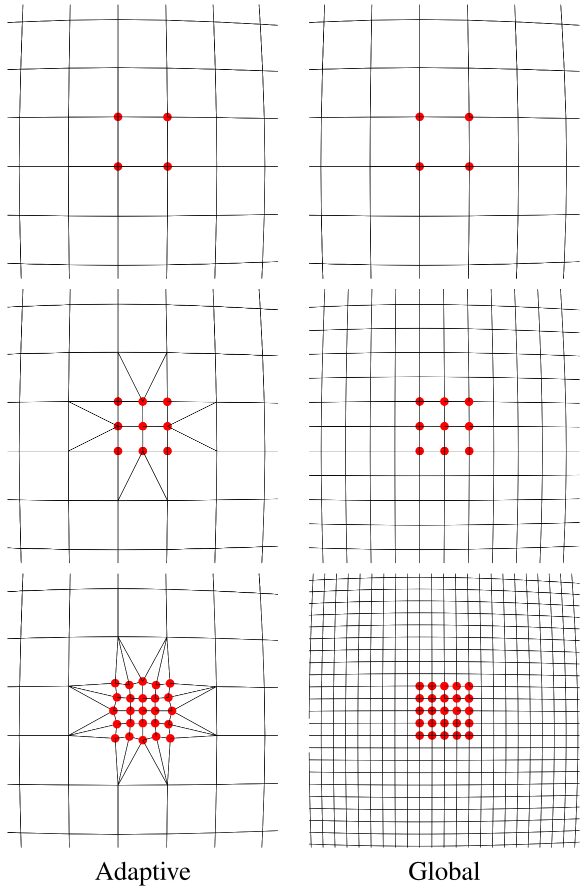

Figure 7. Comparison of adaptive subdivision with simple crack removal algorithm to global subdivision

because quad faces produce ordinary vertices when subdivided. However, if a face with a large number of vertices is subdivided, then this method will lead to many other face refinement. In this case, the first approach is preferred. Usually, faces with more than two cracks are subdivided using the regular Catmull-Clark subdivision rules.

This method removes cracks efficiently, but as illustrated in Figure 7, it has some side-effects. The connectivity and valence of edge-vertices on the cracks are changed. Since these vertices lie within the selected subdivision area, the shape of the limit subdivision surface is modified. Repeated subdivisions of the selected area produces high valence vertices where new edges are introduced to remove cracks. High valence vertices make long faces that are undesired in both analysis and rendering of the surface. Finally, if a selected region is subdivided repeatedly, the subdivision depth difference between the selected and unselected areas becomes larger after each subdivision. This results in a sudden change in the resolution of the mesh from coarse to fine areas and is analogous to aliasing. In rendering and mesh analysis applications, it is desirable to have a smooth transition from the coarse to fine regions of the mesh.

3.3. A Combined Method

Subdivision of a subset of the faces of the mesh changes the connectivity of some vertices within the selected subdivision area, and therefore it affects the resulting limit surface. In adaptive subdivision for triangular meshes, two separate methods attempt to avoid this side effect. One method restricts the mesh by enforcing the vertices involved in the subdivision process to be at the same subdivision depth [13]. If during subdivision it is determined that the subdivision depth of vertices do not match, then the adjacent faces at lower subdivision level are refined until all vertices have the same subdivision depth. The other method, red-green triangulation [2], attempts to avoid irregular shaped meshes. Cracks are removed using CREs, but if a face contains more than one crack per edge, then the face is regularly refined. This effectively limits the subdivision depth difference of adjacent faces. It is possible to combine these two methods to obtain a proper adaptive subdivision surface with desirable connectivity.

Figure 8 shows an example of extending these algorithms to Catmull-Clark subdivision. To restrict the mesh, for each selected vertex the subdivision depth of its neighbours are checked. If they do not match, then the face with the lower subdivision depth is subdivided. During subdivision, CREs are used to remove crack as described previously, and if a face contains more than one crack per edge, then it is regularly refined. Note that, if only quads are produced when removing cracks, then the mesh is always restricted. However, in the general case of meshes with arbitrary faces, the mesh restriction criterion is required.

Although this combined method satisfies all the require-

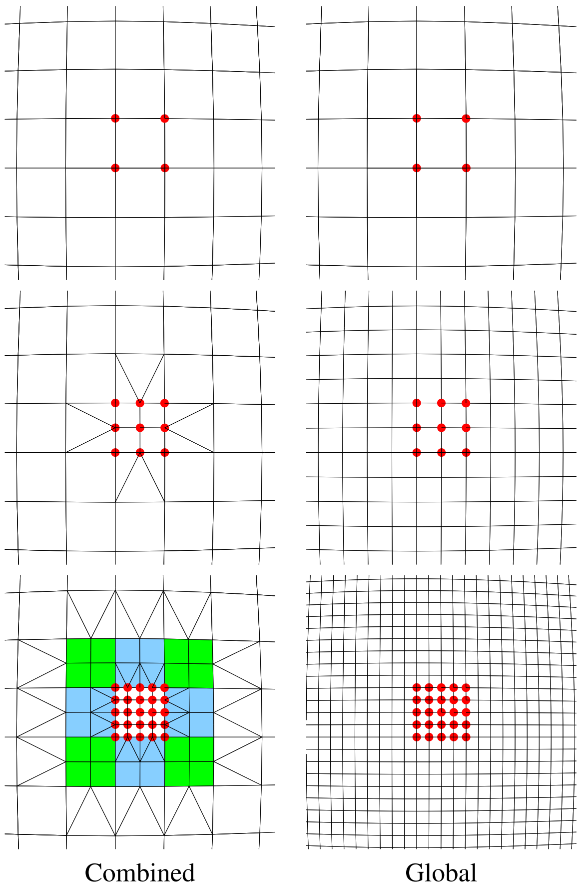

Figure 8. Combination of mesh restriction and red-green triangulation to remove cracks. The blue faces are refined because of cracks, and the green faces are refined to satisfy the restriction criteria.

ments that we set earlier in this paper, it is not efficient. Each face refinement, whether performed to obtain a restricted mesh or to avoid multiple cracks per edge, may produce new cracks. Thus, a single adaptive subdivision step involves recursive checking for cracks and ensuring mesh restriction until no faces are split. In the worst case, this algorithm may refine all faces of the mesh. Incremental subdivision not only produces proper adaptive subdivision surfaces, but also it is not more complex than regular subdivision.

4. Incremental Catmull-Clark Subdivision

We now describe our new incremental method of adaptive subdivision for Catmull-Clark surfaces. This algorithm generates meshes with proper connectivity and geometry. The generated surface gradually changes in resolution from coarse to fine areas. Reflecting the local nature of Catmull-Clark subdivision scheme, our incremental subdivision also operates locally on the mesh.

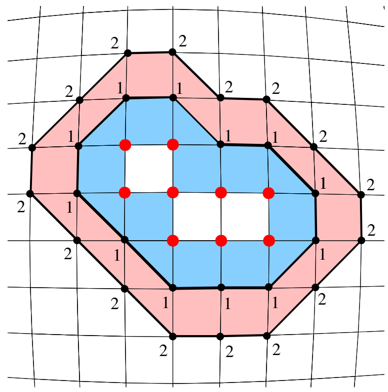

Figure 9. Ring one and two expansion of the selected vertices

Let $V = \{v_0, v_1, \ldots, v_{k-1}\}$ be the vertex set of the current mesh. Let S be subset of V. We wish to adaptively subdivide S such that the limit surface generated from S is exactly the same as when V is globally subdivided. To do this, we expand the selected set S to a new larger set of vertices, and then we subdivide this larger set. More formally, at each subdivision level, expand S to $E^r(S)$ by including the vertices of V that are inside the r-ring neighbourhood of at least one vertex of S

$$E^r(S) = \bigcup_{v \in S} N^r(v), r > 0, \qquad (3)$$

where $N^r(v)$ denotes the r-ring neighbourhood of v. Therefore, $w \in V$ is in $N^r(v)$ if and only if there is a path from v to w with maximum r edges. In graph theory terms, the distance of v and w must be smaller or equal than to r. Figure 9 shows $E^1(S)$ and $E^2(S)$ of a group of selected vertices. Next, subdivide $E^r(S)$ and use CREs to remove cracks. Let $S^{'}$ be the new selected area that is the result of subdividing S. Figure 10 illustrates two steps of incremental subdivision by using $E^1(S)$ expansion. As Figure 11 shows, incremental subdivision can be used whether quads or triangles are used when removing cracks in faces with two cracks.

Adaptive subdivision of $E^r(S)$ produces a limit surface from S that is exactly the same as when the entire mesh is subdivided. The reason for this is that edge-vertices whose connectivity was changed to remove cracks lie outside $E^r(S^{'})$, so the connectivity of vertices within $S^{'}$ remains unchanged. In addition, vertices of $S^{'}$ and their neighbours are at the same subdivision depth because $E^r(S)$ includes the r-ring neighbours of S in the subdivision process. Therefore, the limit surface of S is not changed due

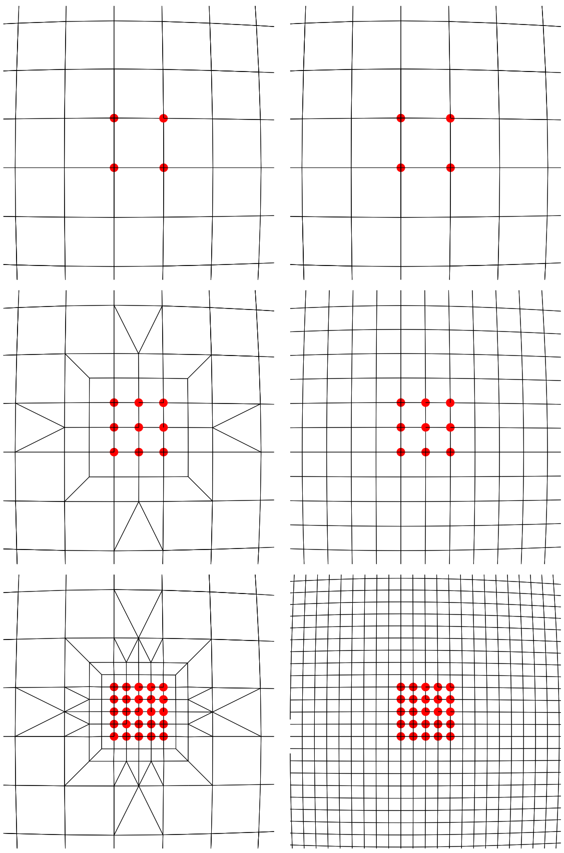

Figure 10. Incremental subdivision on the left. The dots indicate selected vertices for subdivision.

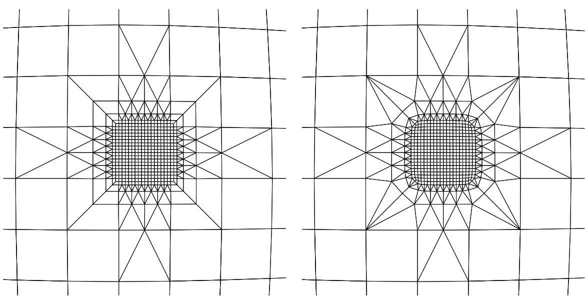

Figure 11. Comparison of the meshes produced when (a) quads are used and (b) triangles are used to remove cracks in faces with two cracks.

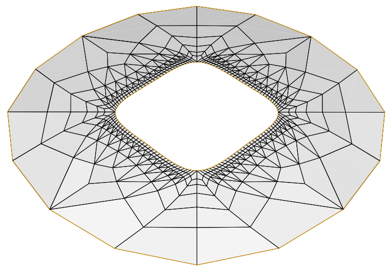

Figure 12. Incremental subdivision of the faces on the interior boundary loop results in a smooth change in the subdivision level from coarse to fine areas

to the adaptive subdivision. Furthermore, an r-ring neighbourhood of S before subdivision corresponds to $2r$-ring neighbourhood of $S^{'}$. Therefore, edge-vertices with modified connectivity to remove cracks and the opposite vertices they are connected to are $2r$ and $3r$ edges away from $S^{'}$, respectively. This prevents the incremental subdivision algorithm from producing high valence vertices when $S^{'}$ is subdivided. Finally, $3r$-ring neighbours of $S^{'}$ are always one level of subdivision lower than its $2r$-ring neighbours. This yields a surface that gradually increases in subdivision depth from coarse regions to the incrementally subdivided areas. Figure 12 demonstrates this effect that is analogous to anti-aliasing. Larger values of r result in surfaces with smoother transition from coarse to fine.

Incremental subdivision of a selected area that includes boundaries is the same as incremental subdivision of interior vertices, with the difference that boundary edges are split according to crease rules.

5. Results

To implement subdivision we used the *vertex-vertex systems* [10] that accurately represents the local nature of subdivision operations. This data structure holds a set of vertices along with an ordered set of their neighbours. A single pass over the mesh creates face-vertices and edge-vertices, repositions vertex-vertices, and then interconnects the new vertices. Incremental subdivision is a simple extension of this algorithm, but operates only on the selected vertices for subdivision. During subdivision, r-ring unselected neighbours of each selected vertex are tagged and included in the subdivision process. Edge-vertices are only created on edges with both vertices selected or tagged. A face-vertex is created when all vertices of the face are selected or tagged. Finally, when the new vertices are interconnected, the cracks are removed by using CREs as described in section 3.2.

Incremental subdivision can be used in a number of ge-

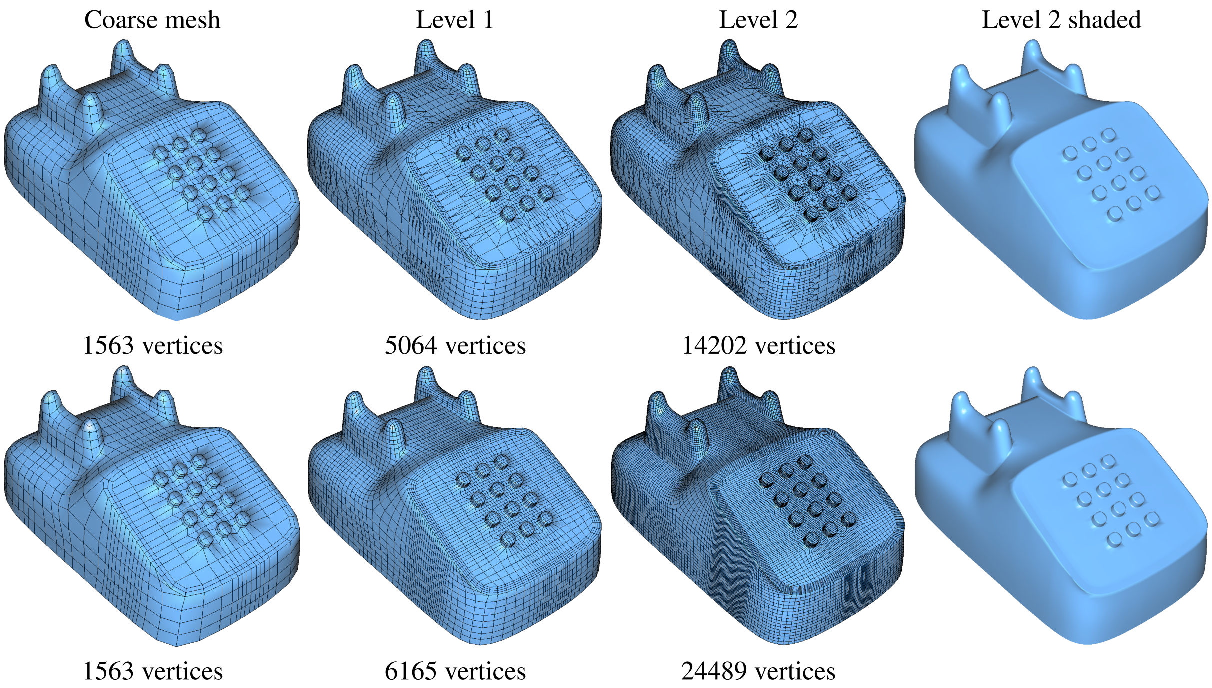

Figure 13. Comparison of incremental subdivision based on dihedral angle (top) and global subdivision (bottom)

ometric modeling applications. An example of such application is creating efficient smooth surfaces by only subdividing high curvature regions of the mesh. In Figure 13, dihedral angle is used to select and incrementally subdivide high curvature areas of the model. A benefit of using incremental subdivision is that subdivision depth of adjacent faces is limited, so the level of detail of the surface changes gradually from coarse to fine areas. Since no analytical precomputation is required, the model can be edited at any subdivision depth. Users may also locally increase the resolution of the mesh for more controlled editing as illustrated previously in Figure 4. Incremental subdivision can also be used to add sharp features to a model. In Figure 14, faces on sharp or boundary edges are incrementally subdivided. An advantage of incremental subdivision is that face selection is performed automatically through $E^r(S)$ expansion at each subdivision level. This method also produces surfaces with fewer face count while achieving the same visual quality as subdivision of the entire mesh.

6. Conclusion

Adaptive subdivision allows us to create surfaces with different subdivision depths by subdividing select areas of the input mesh. To remove cracks–that are the result of subdivision depth difference of adjacent faces–new edges are introduced into the mesh. However, this method of removing cracks produces surfaces with undesirable properties. Using a combined method of restricting the mesh and limiting the depth difference of adjacent faces, it is possible to adjust this method to obtain better behaved adaptive subdivision surfaces. The disadvantage of this algorithm is that it is inefficient. We introduced incremental adaptive subdivision for Catmull-Clark subdivision surfaces. It produces surfaces that have proper connectivity and geometry with a gradual change in subdivision depth from coarse and fine areas. Based on our comparison, incremental adaptive subdivision is more efficient than the combined method to remove cracks, while it is still simple to implement. It can also be effectively used in modeling and rendering applications.

Acknowledgements

We would like to thank Colin Smith for his help and discussion on adaptive subdivision. We also would like to thank Maria Perdomo for proof reading this paper. This research is partially funded by Natural Sciences and Engineering Research Council of Canada (NSERC).

References

[1] Daniel Azuma, Daniel Wood, Brian Curless, Tom Duchamp, David H. Salesin, and Werner Stuetzle. View-dependent refinement of multiresolution meshes with subdivision connectivity. In *Proc. of the 2nd in-*

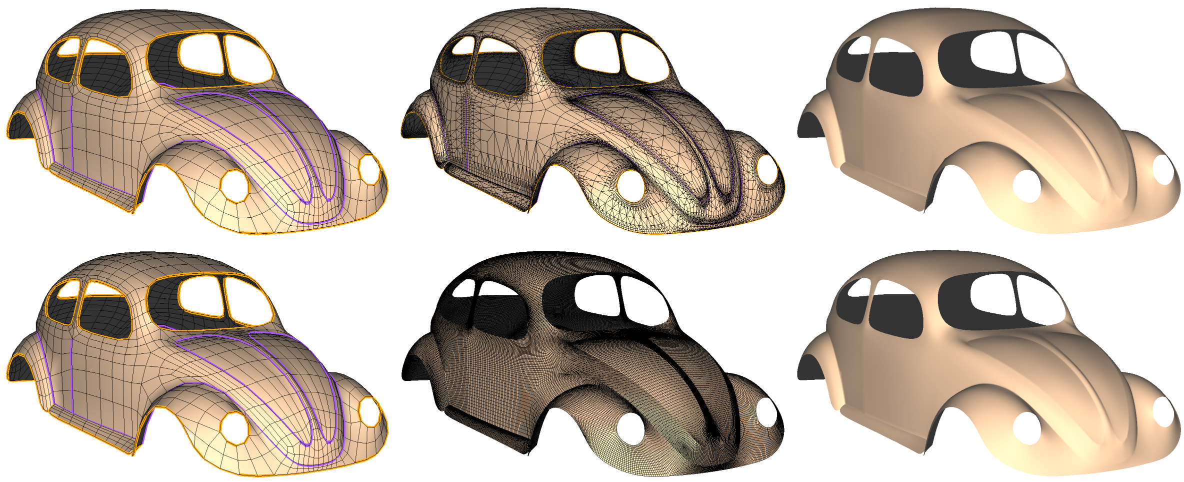

Figure 14. Top row shows incremental subdivision to create sharp features without subdividing the entire mesh. The bottom row shows global subdivision of the mesh to produce the sharp features.

ternational conference on Computer graphics, virtual Reality, visualisation and interaction in Africa, pages 69–78. ACM Press, 2003.

[2] Randolph Bank, Andrew Sherman, and Alan Weiser. Refinement algorithms and data structures for regular local mesh refinement. In *Scientific Computing*, volume 1, pages 3–17. IMACS/North-Holland, 1983.

[3] Henning Biermann, Adi Levin, and Denis Zorin. Piecewise smooth subdivision surfaces with normal control. In *Proc. of the 27th annual conference on Computer graphics and interactive techniques*, pages 113–120. ACM Press/Addison-Wesley Publishing Co., 2000.

[4] Edwin Catmull and James Clark. Recursively generated b-spline surfaces on arbitrary topological meshes. *Computer-Aided Design*, 10(6):350–355, September 1978.

[5] Tony DeRose, Michael Kass, and Tien Truong. Subdivision surfaces in character animation. In *Proc. of the 25th annual conference on Computer graphics and interactive techniques*, volume 32, pages 85–94. ACM Press, 1998.

[6] Tobias Isenberg, Knut Hartmann, and Henry König. Interest value driven adaptive subdivision. In *Simulation und Visualisierung*. SCS European Publishing House, March 2003.

[7] Charles Loop. Smooth subdivision surfaces based on triangles. Master's thesis, University of Utah, August 1987.

[8] Heinrich Müller and Reinhard Jaeschke. Adaptive subdivision curves and surfaces. In *Proc. of the Computer Graphics International 1998*, page 48. IEEE Computer Society, 1998.

[9] Hamid-Reza Pakdel and Faramarz Samavati. Incremental adaptive loop subdivision. In *Proc. of ICCSA'04*, volume 3045 of *LNCS*, pages 237–246. Springer-Verlag, May 2004.

[10] Colin Smith, Przemyslaw Prusinkiewicz, and Faramarz Samavati. Local specification of surface subdivision algorithms. In *Applications of Graph Transformations with Industrial Relevance*, volume 3062 of *LNCS*, pages 313–327. Springer-Verlag, 2004.

[11] Mario Costa Sousa, Kevin Foster, Brian Wyvill, and Faramarz Samavati. Precise ink drawing of 3d models. *Special issue of Computer Graphic Forum*, 22(3):369–379, 2003.

[12] Abhijit Sovakar and Leif Kobbelt. Api design for adaptive subdivision schemes. *Computers & Graphics*, 28(1):67–72, February 2004.

[13] Denis Zorin, Peter Schröder, and Wim Sweldens. Interactive multiresolution mesh editing. In *Proc. of the 24th annual conference on Computer graphics and interactive techniques*, volume 31, pages 259–268. ACM Press/Addison-Wesley Publishing Co., August 1997.