

CLAIMS:

1. A computer-implemented method comprising: generating, by a computer system, a tessellation based on stored domain points, wherein the generating comprises: retrieving pre-computed domain points of an inner tessellation from one of an even tessellation factor table or an odd tessellation factor table; and providing the tessellation for display; making an innermost inner tessellation smaller than an outermost inner tessellation; and calculating inner tessellations at a plurality of edge levels of detail.
2. The method of claim 1, wherein the generating further comprises: converting the domain points to the $[0,1]$ domain.
3. The method of claim 2, wherein the converting comprises: converting the domain points from signed integer format to floating point format.
4. The method of claim 1, wherein the retrieving comprises: retrieving domain points in an order commencing from an inner most domain point of the tessellation to an outer domain point in a spiral-like order.
5. The method of claim 1, wherein the generating further comprises: reading indexes of corner points of triangles from an index buffer.
6. The method of claim 1, wherein the generating further comprises: generating domain points along outer edges of the tessellation.
7. The method of claim 1, further comprising: arranging at least lower factor tessellation within at least one higher factor tessellation; and storing the arranged tessellation in an order commencing from an inner most domain point of the arranged tessellation to an outer domain point in a spiral-like manner.
8. The method of claim 1, further comprising: storing indexes of triangle corners in an index buffer, wherein indexes of corners are associated with domain points.
9. The method of claim 1, wherein the tessellation comprises at least one of a quad or a triangle.
10. A non-transitory computer-readable medium that stores instructions, which when executed by a computer, cause the computer to: generate a patch based on stored domain points, wherein to

generate, the computer is to: retrieve pre-computed encoded domain points of an inner tessellation of the patch from one of an even tessellation factor table or an odd tessellation factor table, make an innermost inner tessellation smaller than an outermost inner tessellation, calculate inner tessellations at a plurality of edge levels of detail, convert encoded domain points to the [0,1] domain, and retrieve indexes of triangle corners; and provide the patch for display.

11. The computer-readable medium of claim 10, wherein to convert encoded domain points to the [0,1] domain, the computer is to scale and bias the encoded domain points.

12. The computer-readable medium of claim 10, wherein to retrieve encoded domain points, the computer is to: retrieve encoded domain points in an order commencing from an inner most encoded domain point of the tessellation to an outer domain point in a spiral-like manner.

13. The computer-readable medium of claim 10, wherein the encoded domain points are stored in an arrangement of at least one lower factor tessellation within at least one higher factor tessellation.

14. The computer-readable medium of claim 10, wherein to generate a patch, the computer is to generate domain points along outer edges of the patch.

15. The computer-readable medium of claim 10, wherein to retrieve indexes of triangle corners, the computer is to retrieve indexes from either a table of even or odd levels of detail.

16. An apparatus comprising: a processor unit to generate a patch, wherein to generate a patch, the unit is to retrieve domain points of a patch, wherein to retrieve domain points, the tessellator is to retrieve pre-computed domain points of an inner tessellation from one of an odd table or an even table and said processor to make an innermost tessellation smaller than an outermost inner tessellation and calculate inner tessellations at a plurality of edge levels of detail; and a domain shader to convert domain points to [0,1] domain and to generate a three dimensional point on the patch.

17. The apparatus of claim 16, wherein to convert domain points to the [0,1] domain, the domain shader is to scale and bias the encoded domain points.

18. The apparatus of claim 16, wherein to retrieve domain points, the tessellator is to: retrieve domain points in an order commencing from an inner most domain point of the inner tessellation to an outer domain point of the inner tessellation in a spiral-like manner.

19. The apparatus of claim 16, wherein the domain points are stored in an arrangement of at least one lower factor tessellation within at least one higher factor tessellation.
20. The apparatus of claim 16, wherein to generate a patch, the tessellator is to generate domain points along outer edges of the patch.
21. The apparatus of claim 16, wherein to generate the patch, the tessellator is also to retrieve indexes of triangle corners from a table of odd or even levels of details.
22. A system comprising: a display; and a computing system to generate patches to transmit to the display, wherein the computing system comprises: a tessellator to generate a patch, wherein to generate a patch, the tessellator is to retrieve domain points of a patch, wherein to retrieve domain points, the tessellator is to retrieve pre-computed domain points of an inner tessellation from one of an odd table or an even table, the tessellator to make an innermost inner tessellation smaller than an outermost inner tessellation and calculate inner tessellations at a plurality of edge levels of detail; and a domain shader to convert domain points to a $[0,1]$ domain and to generate a three dimensional point on the patch.
23. The system of claim 22, wherein to convert domain points to the $[0,1]$ domain, the domain shader is to scale and bias the encoded domain points.
24. The system of claim 22, wherein to retrieve domain points, the tessellator is to: retrieve domain points in an order commencing from an inner most domain point of the inner tessellation to an outer domain point of the inner tessellation in a spiral-like manner.
25. The system of claim 22, wherein to generate the patch, the tessellator is also to retrieve indexes of triangle corners from either a table or odd or even levels of details.