平成20年度図書館情報メディア研究科プロジェクト研究　研究成果報告書

種　　目	萌芽研究		研究代表者 氏　　　名	金　尚泰
研究課題	３次元コンピュータグラフィックスにおけるポリゴン間引きの自動化に関する研究			

研究組織（研究代表者及び研究分担者）

氏　　名	所属研究機関・部局・職	現在の専門	役割分担
金　尚泰	図書館情報ﾒﾃﾞｨｱ研究科・開発分野・講師	グラフィックデザイン	個人研究

研究目的

ポリゴンで構成される３D コンピューターグラフィックスデータ（以下３DCG）は、空間を表示するデータの仕組みとともに短い直線の固まりで曲線を表現するため、計算機内部で処理する情報量は大きい。そのデータ上の特性により、３D モデルデータの活用に関する難しさは、多少改善されではいるが、現在も昔も殆ど変わらない。３DCG をコンテンツとして扱うためには、大容量のメモリー、磁気媒体や高性能グラフィックボード、さらに CPU の処理能力などが重要な変数になることで一般ユーザには、扱いにくい分野でもある。本研究の目的は、3 次元 CG で制作されたポリゴンデータの最適化にインテリジェンスを与えることによって、データ容量を多いに減らすことの出来る新しいアルゴリズムのリサーチと、その試作プログラムを試みる。細かく表現されているモデルデータでも、最終的にコンテンツとしてまとめるには、データ容量をより効率よく減らし、手作業での修正を再度しなくてはならないのが現実である。さらにバーチャルリアリティやウェブ３D コンテンツなど、テクノロジーと融合、もしくはリアルタイムでの操作による利用は、モデルの外形をまったく崩さずにデータ容量を減らすには多大な費用と手間がかかるのも現状である。（＊別紙参照）このようなインテリジェンス（形を認識し自動で間引く）化されたプログラムは、これからの研究としても、社会貢献としても多方面に評価のできる重要なテーマとして考えている。

研究成果

圧縮アルゴリズム体系確立（**3D Polygon Compression Algorithm**）

Compression will work on triangulated data. Compression follows the algorithm of Vertex Merge (also known as Edge Collapse).

1. Load the model’s triangle face data and create Adjacency structures.
a) Load all the vertices of the model into an array: vertexArray.
b) Load all the faces of the model into an array: faceArray.
c)After loading the faces, create a data structure such that each vertex is aware of all its connected vertices.
d)Create a data structure such that each vertex is aware of all of its connected faces.

2.Detect Boundary Vertices / Edges
a)Find all edges that are members of only one triangular face.
b) These are called Boundary Edges.
c)Mark all the vertices that are members of these Boundary Edges as Boundary Vertices.(Compression takes care to give the Boundary Vertices higher importance) The Red vertices are part of boundary edges. Blue vertices are not boundary vertices.

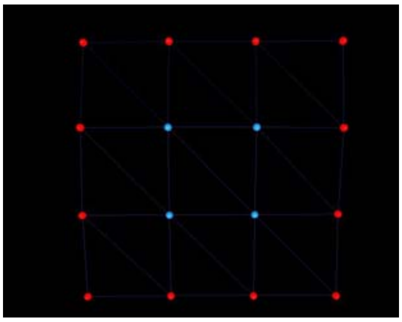


Fig.2 Vertices Highlighted (Boundary Vertices in RED)

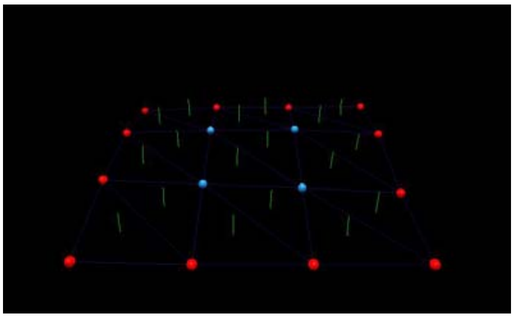


Fig.3 Surface Normal of the Mesh in Green

3.Find Edge Importance for ALL edges in the model and then find Vertex Impact for each vertex.
a)Take each vertex and loop through all connected vertices.
b)Find all faces that share this edge.
c)Calculate the face that is oriented in a direction furthest away from the rest of the faces connected to the current vertex.
d)The angle between faces is found by the vector dot product of their Surface normal vectors.
e)The min dot product value is taken as the “Impact” of this vertex on the model. The vertex impact is therefore based on the local curvature with neighboring faces.
f)For Boundary vertices, we add a constant ‘B’ to its Impact. Thus the Impact or importance of this vertex is increased.
g)After looping through all the connected vertices, we see that each vertex pair has an impact. Find the vertex pair that have the least impact. The second vertex in this pair is the Merge destination vertex for the current vertex. So if this vertex was chosen to be collapsed or merged, then it would be merged to its Merge Destination vertex.
h)Each vertex should store the Impact value and its Merge Destination vertex.i)Got to step a. and repeat for all vertices.

4.Sort the entire vertex list in descending order of Vertex Impact.
a)The top of the list has vertices with maximum Impact or importance. These directly correspond to vertices with higher local curvature.
b)The Bottom of the list has vertices with minimum Impact or importance.

5.Perform Vertex Merge
a)Select the vertex at the bottom of the sorted vertex list.
b)This represents the vertex that is least important and contributes the least to the appearance of the model. (ie. Removing it makes the least change in the visual quality of the model).
c)Let this vertex be called “vA”.
d)Find the Merge Destination Vertex of “vA” . We will call it vMD.
e)We are now going to delete vertex vA.
f)But first we need to inform all faces attached to vA that they should now connect with vertex vMD instead of vertex vA.
g)The Surface normal vectors for all these changed faces have to be recalculated.
h)All vertices involved in the operation will now have different Impacts. So using Step 3, recalculate all the vertex Impacts for a all these vertices.
i)Got to Step 4 and sort the entire vertex list again.
j)Then got Step 5 and perform Vertex merge on the next least important vertex.
k)With each Vertex Merge:1 Edge is deleted, 1 Vertex is deleted, 2 Faces are deleted. This is the fundamental idea behind compression. Continue this process until we have compressed to desired polygon count.

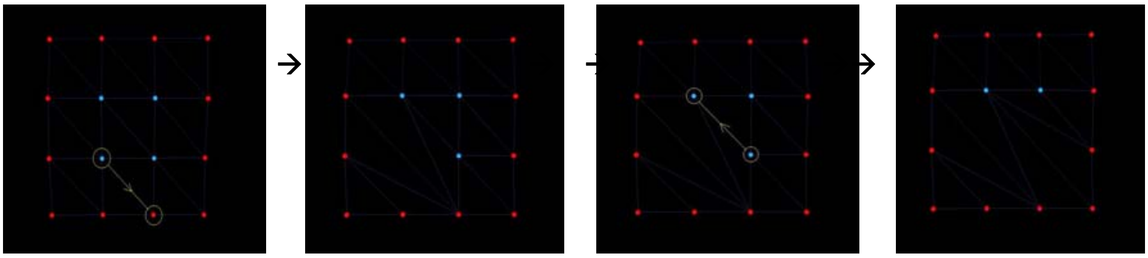


Fig.4 An Example of Vertex Collapse

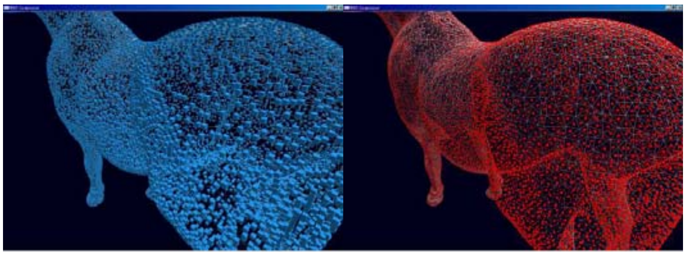


Fig.5 left: Center of Vertex Normal, right: Center of Faces Normal

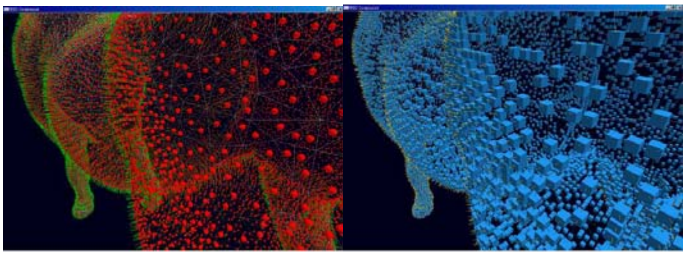


Fig.6 left: Vertex Normal, right: Faces Normal

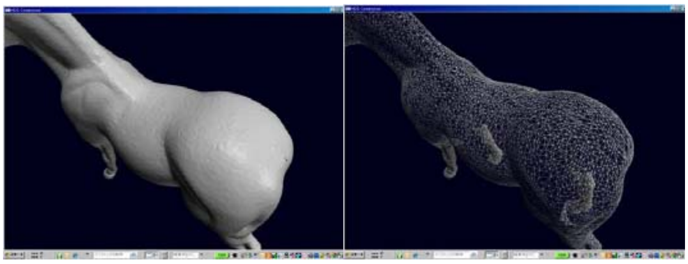
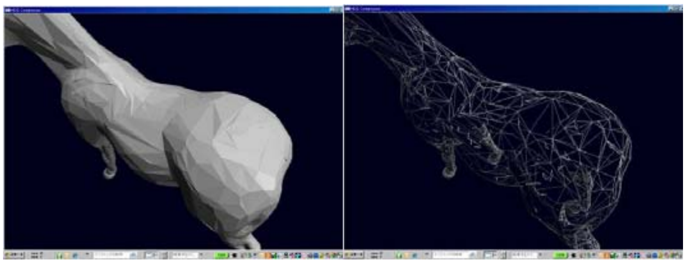
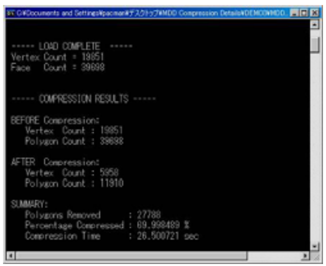


Fig.7 Comparison of Before and After Compression





代表的な研究発表・特許等の成果一覧、特記事項等

Sangtae, KIM, YAMANE Chikako、A Development and Application of a New Generation of Graphic Technology、International Service Innovation Design Conference, Pusan Korea2008, pp314-322, SIDC 2008.