



Special Module on Media Processing and Communication

Dayalbagh Educational Institute
(DEI)
Dayalbagh Agra

CSM 011

Indian Institute of Technology Delhi
(IITD)
New Delhi

SIV 864

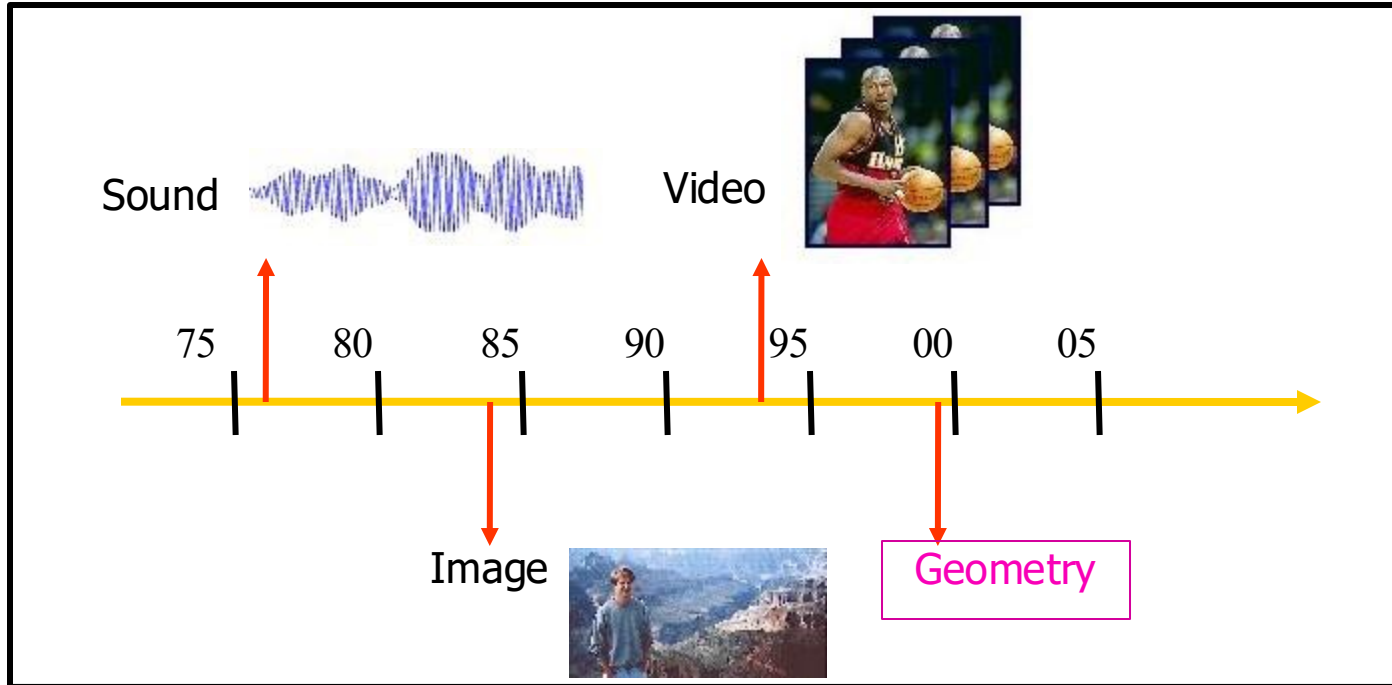


Geometry Compression



Multimedia

Historical Perspective: Digital Media



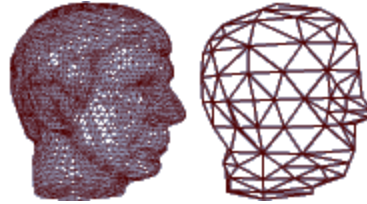


Geometric Models

Representations



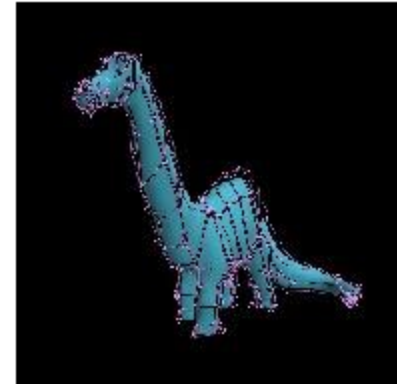
Constructed Solid
Geometry



Polygonal Surface



Implicit Surface



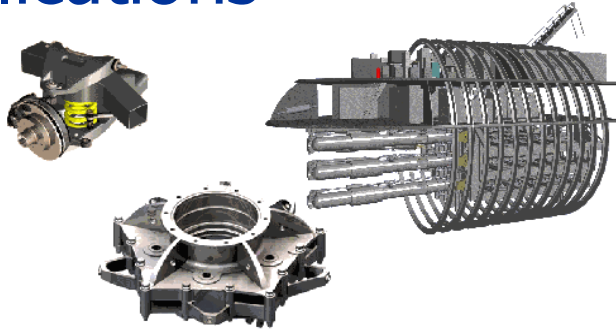
Parametric Surface



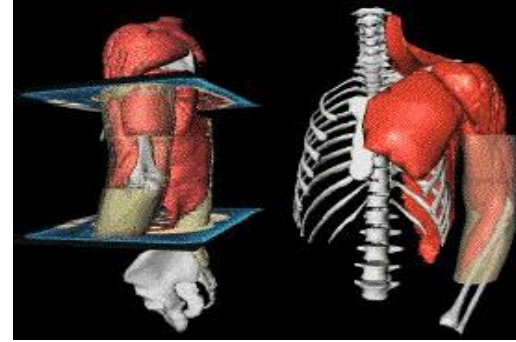
Geometric Models

Applications

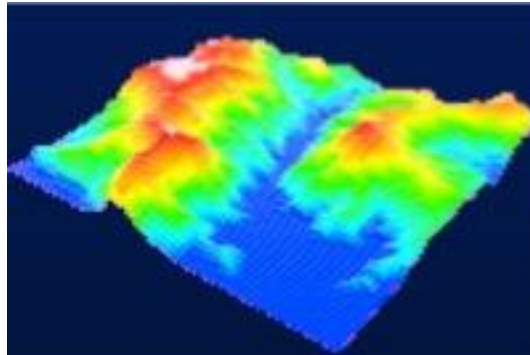
Engineering



Medical



Topography



Art

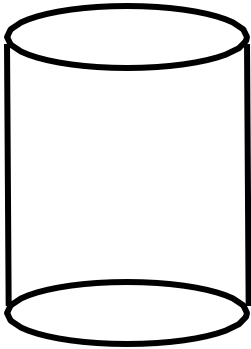




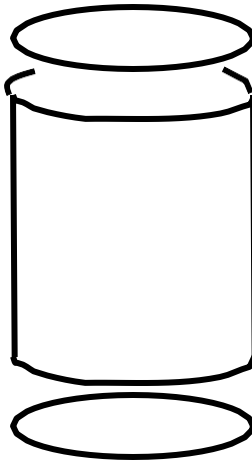
Geometric Models

Polygonal Representation

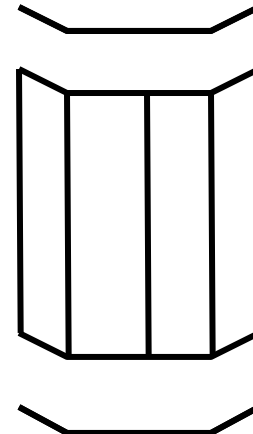
For rendering often object is represented as collection of polygons



Object



Surfaces



Polygonal Patches



Geometric Models

Polygonal Representation

Polygonal mesh is a collection of edges, vertices and polygons such that each edge is shared by at most two polygons

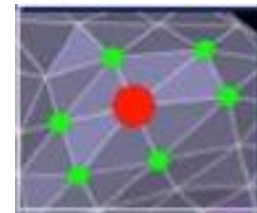
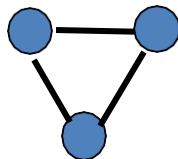
Vertex: Node



Edge: Connects two vertices



Polygon: Closed sequence of edges





Geometric Models

Polygonal Representation

Explicit Representation

Each polygon is represented by

$$P = ((x_1, y_1, z_1) (x_2, y_2, z_2) \dots (x_n, y_n, z_n))$$

i.e. vertices are stored in the order of traversal

Edges connect the successive vertices plus the last one

This representation has restrictive manipulation and has multiple storage of points.



Geometric Models

Polygonal Representation

Pointer to Vertex List

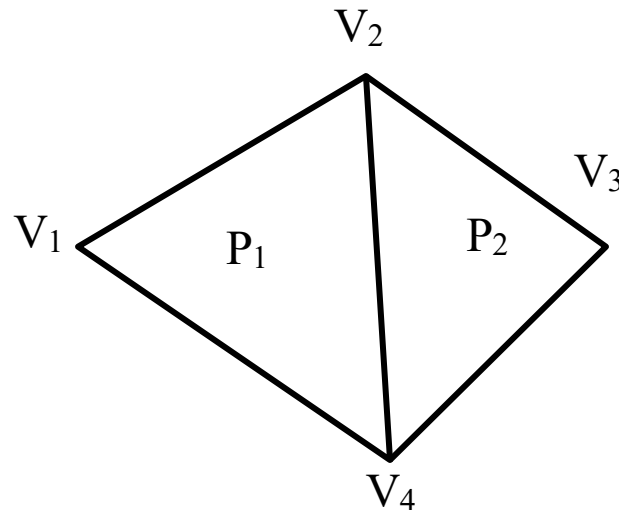
Each vertex is stored once in a list V

$V = ((x_1, y_1, z_1) (x_2, y_2, z_2) \dots (x_n, y_n, z_n))$

Each polygon is represented as

$P = (V_1, V_2, V_3)$

e.g. $P_1 = (1, 2, 4)$ and $P_2 = (4, 2, 3)$



In this representation it is difficult to find polygons that share an edge.



Geometric Models

Polygonal Representation

Pointer to Edge List

Edge: $E = (V_i, V_j, P_m, P_n)$

Polygon = (E_p, E_q, E_r)

$V = (V_1, V_2, V_3, V_4)$

$E_1 = (V_1, V_2, P_1, \text{null})$

$E_2 = (V_2, V_3, P_2, \text{null})$

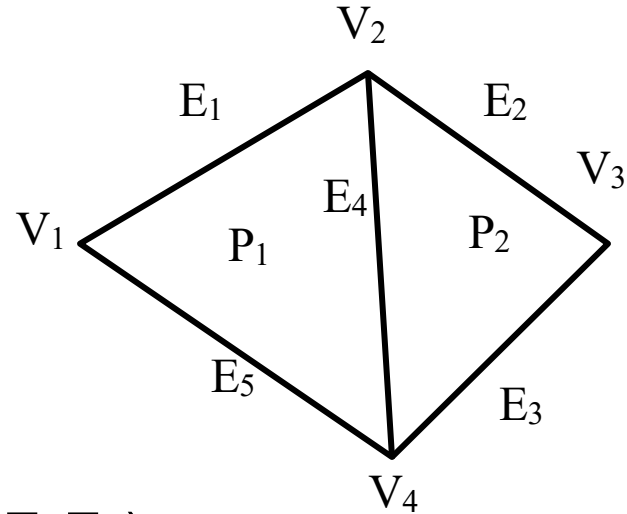
$E_3 = (V_3, V_4, P_2, \text{null})$

$E_4 = (V_4, V_2, P_1, P_2)$

$E_5 = (V_4, V_1, P_1, \text{null})$

$P_1 = (E_1 E_4 E_5)$

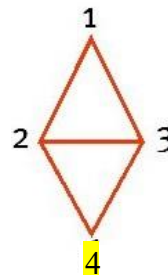
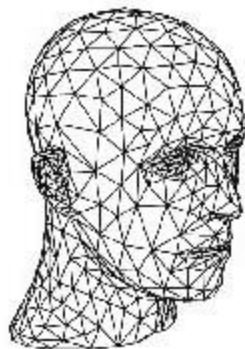
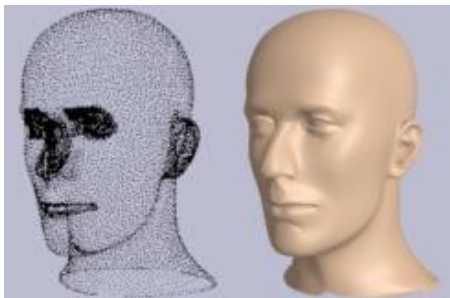
$P_2 = (E_2 E_3 E_4)$





Geometric Models

- Typically,
- Geometry Data: Meshes
 - Points
 - Connectivity



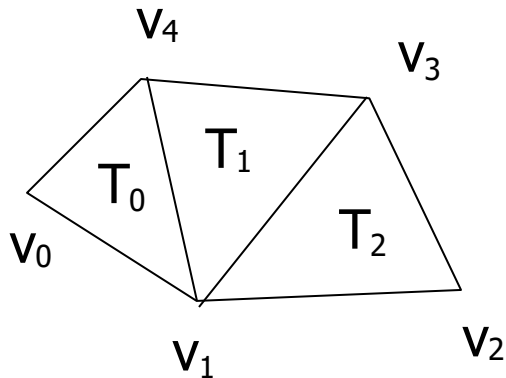
x_1	y_1	z_1
x_2	y_2	z_2
x_3	y_3	z_3
$\vdots$		
1	2	3
2	3	4
$\vdots$		



Geometric Compression

Connectivity Coding

Indexed Face Set



$T_0 : v_0v_1v_4$

$T_1 : v_1v_4v_3$

$T_2 : v_1v_2v_3$

Simple approach

Vertex index of 32 bits
 $3 \times 32 = 96$ bits/triangle

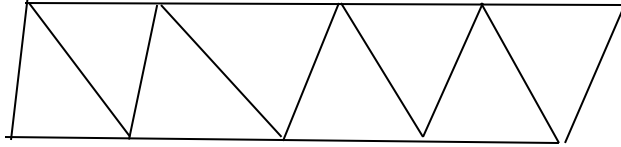
$3 \times \log_2(V)$ bits/triangle
 $= 50$ bits/triangle
(for 100k vertices)



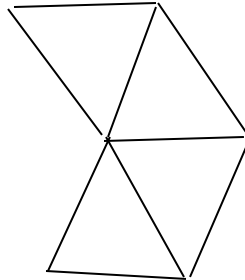
Geometric Compression

Connectivity Coding

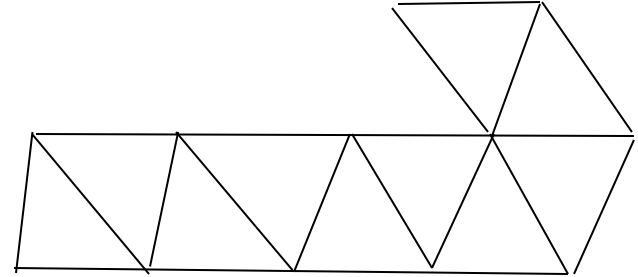
Triangle strip



Triangle fan



Generalized triangle strip

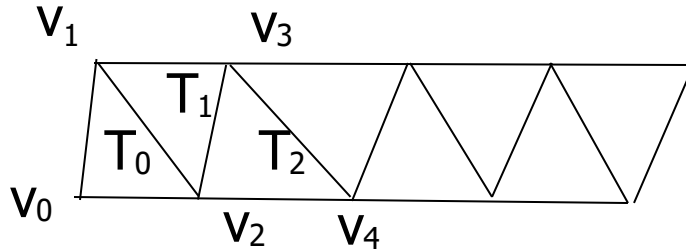




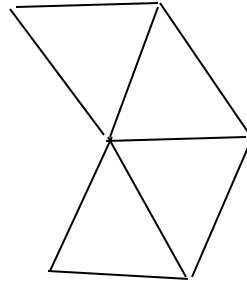
Geometric Compression

Connectivity Coding

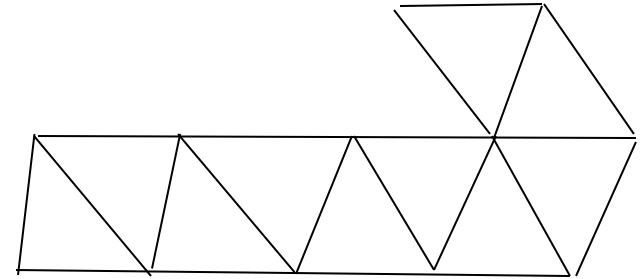
Triangle strip



Triangle fan



Generalized triangle strip

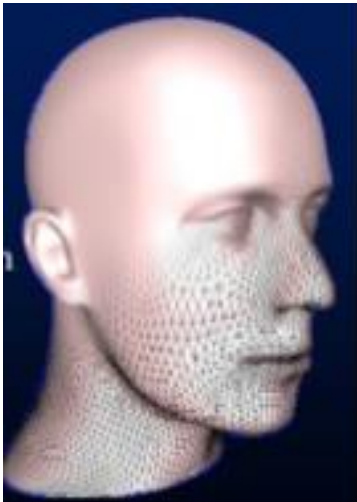


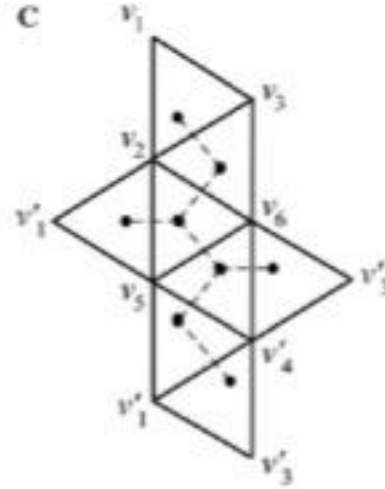
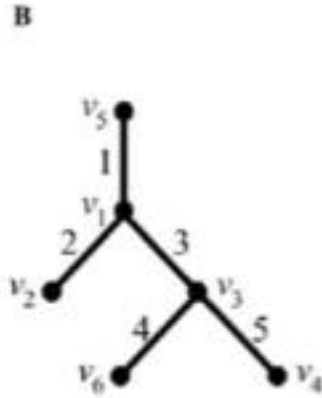


Geometric Compression

Connectivity Coding

Triangle strips



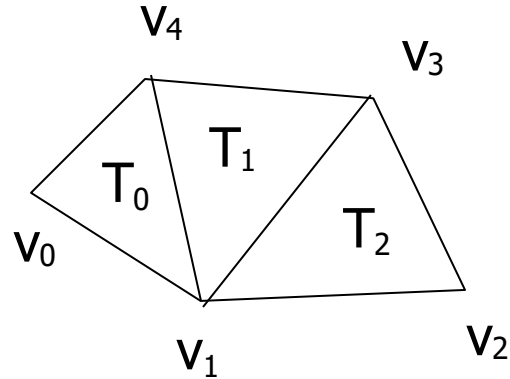




Geometric Compression

Geometry Coding

Scalar quantization



$V_0 : x_0 y_0 z_0$

$V_1 : x_1 y_1 z_1$

$V_2 : x_2 y_2 z_2$

⋮

Simple approach

Each coordinate of 32 bits
 $3 \times 32 = 96$ bits/vertex

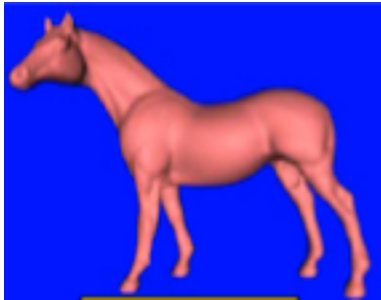
Quantization: 10 bits
 $= 3 \times 10$ bits/vertex



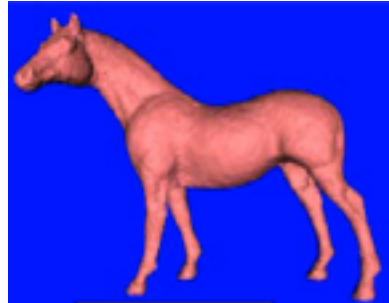
Geometric Compression

Geometry Coding

Uniform quantization



12 bits



8 bits



6 bits

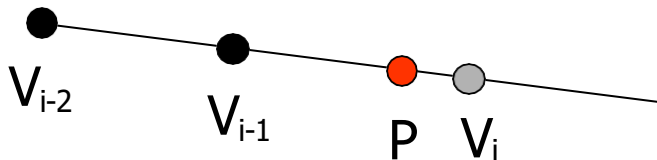


Geometric Compression

Geometry Coding

Prediction: Exploits correlation between adjacent vertex coordinates

Linear prediction: Linear combination of K previous vertices



$$P = V_{i-1} + (V_{i-1} - V_{i-2})$$

$$\text{error } e_i = V_i - P$$



Geometric Compression

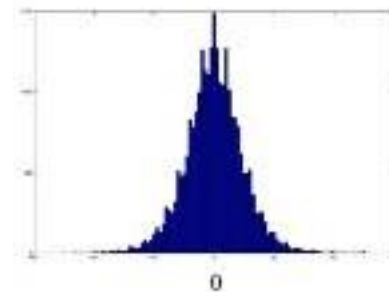
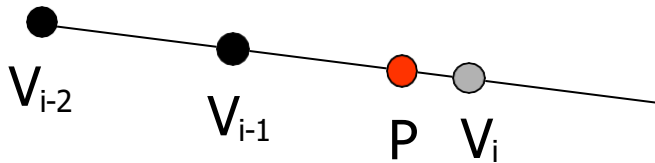
Geometry Coding

Vertex Coordinates: $V_1 V_2 V_3 V_4 \dots$

Using prediction error: $V_1 V_2 e_3 e_4 \dots$

Decoding: $V_1 V_2$

$$V_i = P + e_i \text{ where } P = V_{i-1} + (V_{i-1} - V_{i-2})$$

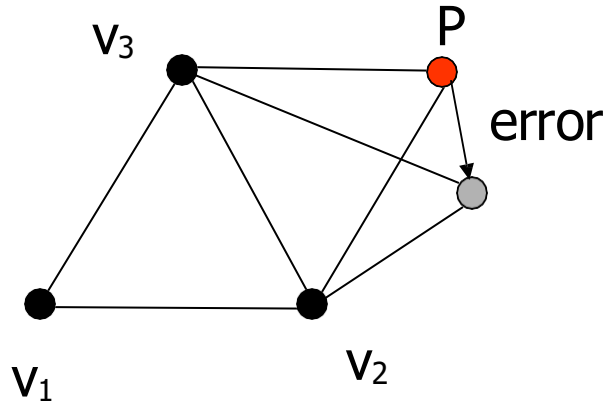




Geometric Compression

Geometry Coding

Parralelogram prediction:



$$V_p = V_2 + V_3 - V_1$$

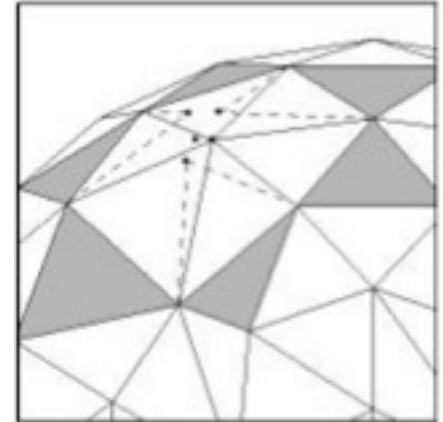
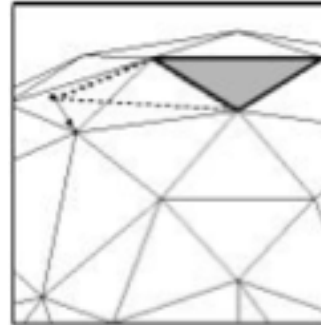


Geometric Compression

Geometry Coding

K-way prediction:

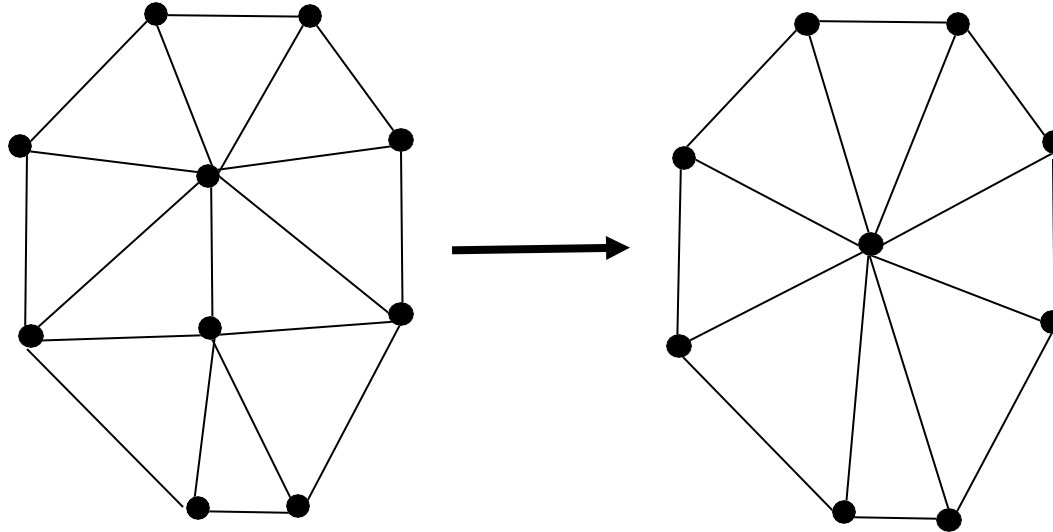
Vertex is average
of its neighbours





Geometric Compression

Geometry Coding Downsampling

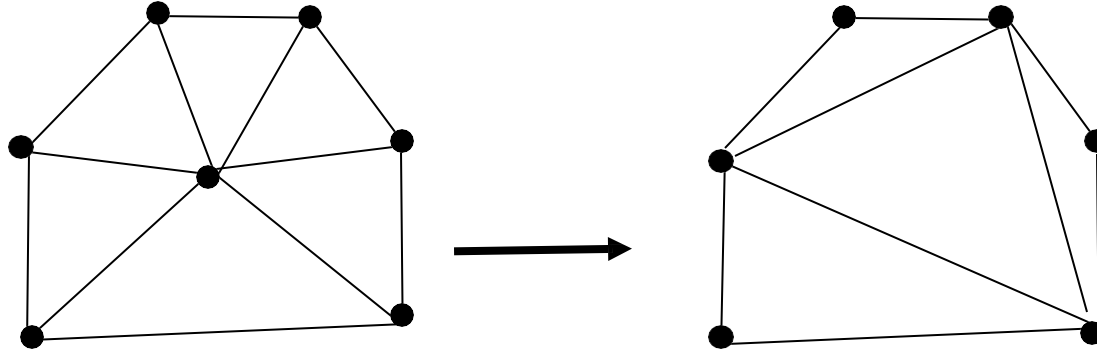




Geometric Compression

Geometry Coding

Downsampling





Geometric Compression

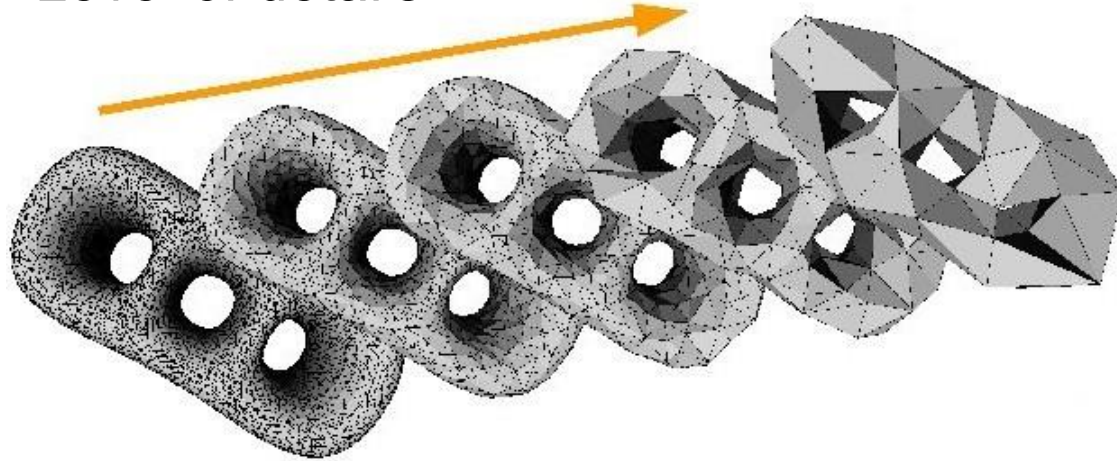
Geometry Coding

Downsampling

Resolution

Level of details

Mesh

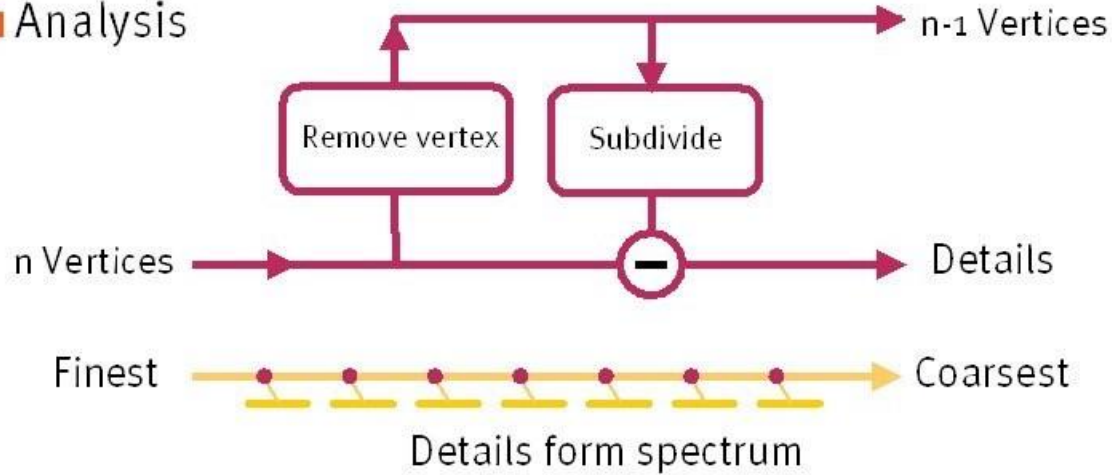




Geometric Compression

Multi-resolution Representation

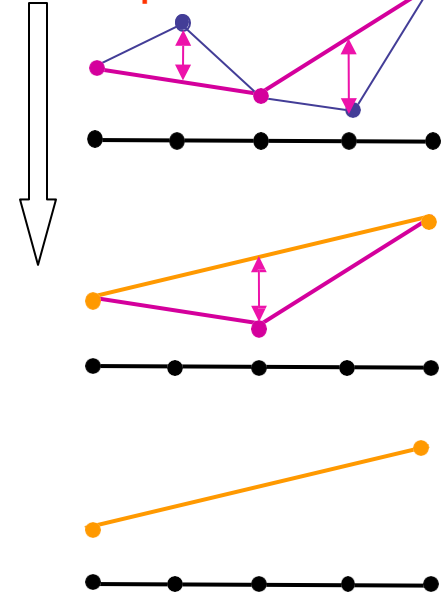
Analysis



Synthesis



Down sample



Wavelet

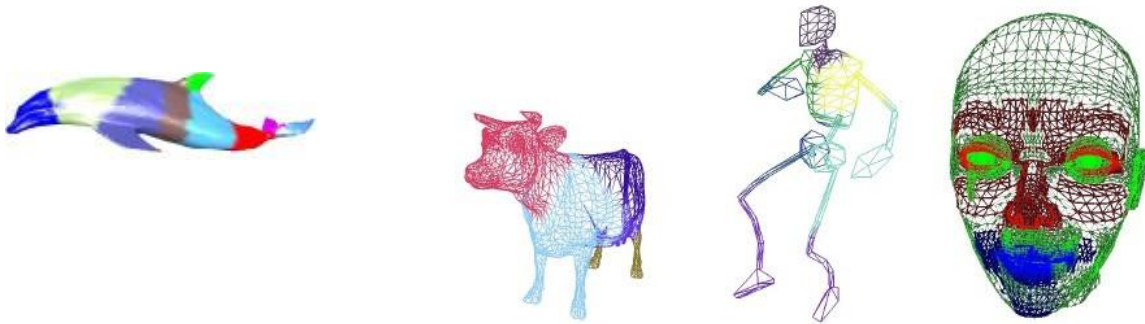
Subdivide



Geometric Compression

Animated Mesh Coding

- Animation can be modeled as a transformation
- Partitioning or clustering of data



- Prediction based method for vertices which move
- Measure to preserve time variant geometry features



Geometric Compression

► References

- Technologies for 3D mesh compression: A survey by J Peng, CS Kim, CC Jay-Kuo, Journal of Visual Communication Image Representation, 16, 2005, 688-733
- <http://imsc.usc.edu/research/project/digitalgeometry/>
- <http://www.cs.ubc.ca/~sheffa/dgp/ppts/Geometry.pdf>
- 3D mesh compression: survey, comparisons and emerging trends, ACM Computing Surveys, Vol 9, No 4, Article 39, September 2013.