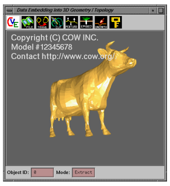
3D Models as Multimedia Data
- Compression and Digital watermarking -

Ryutarou Ohbuchi
Yamanashi University
Computer Science Department

© R. Ohbuchi

3D models are multimedia data

- VRML
  - 3D contents on the web.
- PlayStation 2, Dolphin … (and , X-box?)
  - Polygon games, …
- MPEG 4
  - "Polygon broadcasting"
  - "Polygon phone".
- Movie, advertisement, …



3D geometric CAD data too!

- 3D geometric CAD data usage
  - Collaborative design, concurrent design.
  - Design lifecycle, design reuse.
  - Virtual components.
  - Documentation, parts catalogue.
  - Engineering service, training.
- Keywords
  - Fusion of multiple sources, multiple data types.
  - Communication, accumulation.
  - Sharing, distribution.

CAD data are multimedia data!

Media data types

- Text
- Audio
- Image
- Movie
- 2D geometric model
  - SVG, digital map, etc.
- 3D geometric model
  - Non-CAD models
    - VRML, MPEG4, PlayStation2, etc.
  - CAD models
- Program
  - Java applet, etc.
- Others.

Entertainment, commerce, broadcasting …

3D models are not just for CAD.

A set of operations for a media data type

Input
Output
Editing
Compression
Indexing
Hyperlinking
Authentication
Tamper detection
Security-related operations
Intellectual property management
Similarity-based search
…

A set of operations for a media data type (e.g., 3D CAD data)

© R. Ohbuchi

- We know these operations well for the text data type.
- We know some about 2D image, 2D movie and audio data types.

How about 3D models?

A 3D geometric model

- Shape
  - Polygons, volume, parametric surfaces, etc.
  - We must have this!
- Shape attributes
  - Colour, normal vector, texture coordinates, transparency, etc.
  - Temperature, stiffness, etc.
- Temporal variations of the above
  - Time sequences of coordinates, rotation angles, topology, parameter values, etc.
    - Ex. VRML coordinate/angle interpolators, MPEG 4 SNHC face animation parameters.

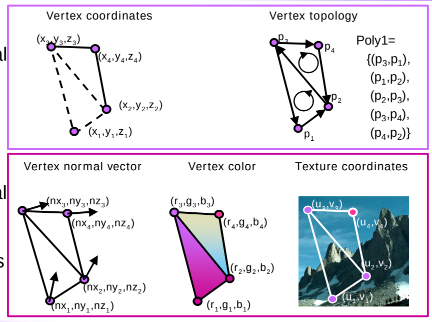
3D shape representations

- Boundary Representation
  - Polygons (triangles, quadrilaterals, etc.)
  - Parametric surface patches.
- Volume Decomposition
  - Lines, contour stacks.
  - (Regular- and irregular-) voxels, octree.
  - Tetrahedrons.
- Procedural Volume Decomposition
  - CSG.
  - Sweep, Minkowski sum, etc.

Shape and Shape Attributes in Polygonal Meshes

Polygonal mesh shape

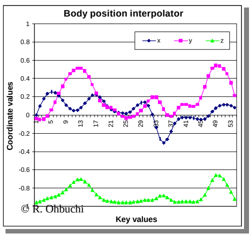
Polygonal mesh shape attributes



Dynamic (time-dependent) 3D models

- MPEG 4 animation param's.
  - Face/body animation param's.
- MPEG 4 Delaunay mesh
  - Mesh topology changes according to geometry of a set of given vertices.
- VRML interpolators
  - Coordinates and angle interpolators.
- "Holoflick"
  - Replace an entire mesh at a video frame rate [Deering99].
    - A 3D movie!

Example: VRML coordinate interpolator.



Compression

- Traditional method
  - Vertex coordinate : 3 × 32bit floating point number.
  - Vertex connectivity: 1 × 32bit × (valence) pointer.
  - Vertex normal vector: 3 × 32bit floating number.
  - Vertex color: 3 × 32bit floating number.
  - Texture coordinate: 2 × 32bit floating point number.
- Total # bytes per vertex
  - 12 + 20 + 12 + 12 + 8 = 64Byte/vertex assuming vertex valence of 5.
- Now, Boeing 777 design contains 1 billion polygons
  - Not including rivets.

We need compression.

Security related operations for media

How do you know the CAD data you just received over the Internet is not tampered with?

–Watch for ".txt.vbs" extension or "FW" headline?

- Security related operations
  - Authentication, tamper detection, IP protection, copyright protection, …
- Technology
  - Digital signature, cryptography, message hash function, digital watermarking, …

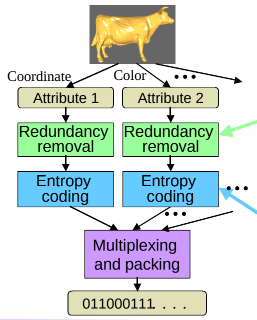
No mechanism is foolproof.

Compression

Outline

- 3D model as multimedia datatype
- **Compression**
- Digital watermarking

Compression?



Lossy
- Bounding box + Quantization
- Polygon simplification
- Etc.

Loss-less
- Coordinate value prediction
- Connectivity encoding by vertex and triangle trees
- Etc.

- Huffman coding
- Arithmetic coding
- Lempel-Ziv77 coding
- Etc.

2000/06/23 Yamanashi University Ryutarou Ohbuchi 13

3D model compression comparison criteria (1)

- Quality loss
  - Loss-less v.s. lossy
    - Loss-less compression is able to recover original data.
- Spatial granularity
  - Single-batch v.s. progressive encoding/decoding
    - Progressive compression encode/decode in small chunks.
    - Single-batch compression has granule size of the entire data.
- Temporal granularity
  - Instantaneously and incrementally encode/decode dynamic (time-dependent) data, i.e., *streaming*.

2000/06/23 Yamanashi University Ryutarou Ohbuchi 14

3D model compression comparison criteria (2)

- Representation power
  - Surface may be open/closed, simple/high-genus, orientable/non-orientable, and manifold/non-manifold.
- Topology constraints
  - If and how strictly the method preserves topology.
- Compression efficiency
  - Required storage capacity or communication bandwidth.
- Attribute mapping
  - If and how strictly the method preserves correspondence between attributes and geometry/topology.

2000/06/23 Yamanashi University Ryutarou Ohbuchi 15

3D model shape compression

- Non-progressive
  - Polygonal mesh (manifold)
    - Deering95 (Java3D), Taubin98a (MPEG4), Touma98, Gumhold98, Rossignac99a, etc.
  - Tetrahedral mesh
    - Szymczak99, Gumhold99
  - Non-manifolds, etc. (“polygon soup”)
    - Masuda98, Rossignac99, Gueziec99, Bajaj99

2000/06/23 Yamanashi University Ryutarou Ohbuchi 16

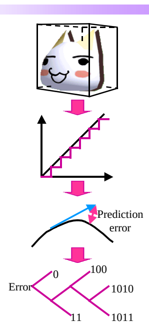
3D model shape compression

- Progressive
  - Polygonal mesh
    - Li98, Taubin98b, Pajarola00, Cohen-Or99.
  - Tetrahedral mesh
    - Pajarola99
- Streaming
  - Immediate and incremental encoding/decoding of dynamically changing data.
    - None!! ⇐ Your name here!

2000/06/23 Yamanashi University Ryutarou Ohbuchi 17

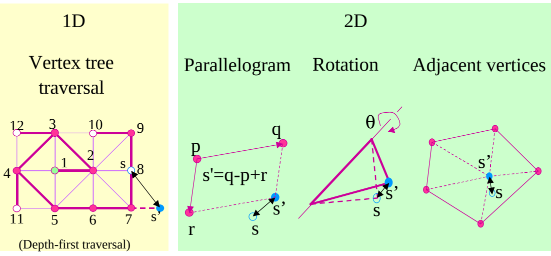
Compressing geometry

- Redundancy removal
  - Normalize by axis-aligned bounding box.
- Quantization
  - 10-12bits/coordinate is enough for viewing.
  - Full resolution for CAD/CAM data?
- Prediction
  - Vertex coordinate prediction
    - 1D: Taubin98a, Taubin98b, etc.
    - 2D: Touma98, Li98, Cohen-Or99.
- Encode prediction error
  - Generate error symbol stream
- Entropy code the symbol stream
  - Huffman code, arithmetic code, etc.



2000/06/23 Yamanashi University Ryutarou Ohbuchi 18

Geometry prediction methods



Encode prediction error=(s-s')

Compressing geometry using 1D coherence

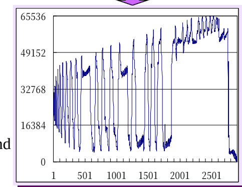
- Create 1D ordering of vertices
  - Depth-first traversal of a vertex tree.
- Apply 1D series prediction
  - Linear Predictive Coding (LPC)

$$\hat{s}(t) = \sum_{i=1}^{n} a_i s(t-i), \quad e(t) = s(t) - \hat{s}(t)$$

$s(k)$ Previous n samples
$\hat{s}(t)$ Next sample (predicted)
$e(t)$ Residual error.
a_i LPC coefficients.

- Encode the residual error
  - Presumably with smaller amplitude, and thus requires less number of bits.

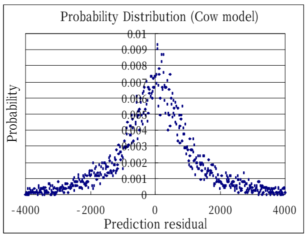
Cow model, 16bit quantization



Compressing geometry using 1D coherence

- Residual error often becomes Laplacian distribution.
  - Small number dominates.
  - Amenable to various entropy coding algorithms.
    - E.g., Huffman coding.

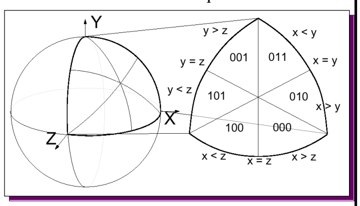
$$p(x) = \frac{1}{\sqrt{2}s} e^{\frac{-\sqrt{2}}{s}|x|}$$



Compressing normal vectors [Deering95]

- Gouraud shaded models not sensitive to normal vectors.
  - Coarse quantization is enough.
  - Normal vector has 2 DOF ($x^2+y^2+z^2=1$)
- Java3D
  - [Deering95]
- MPEG4
  - Similar to Java3D partition.

Java3D normal quantization

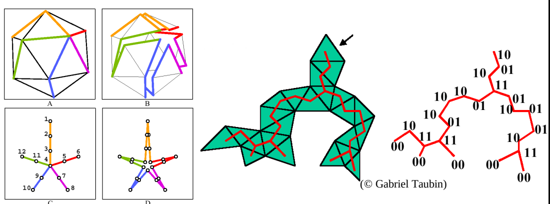


Connectivity compression

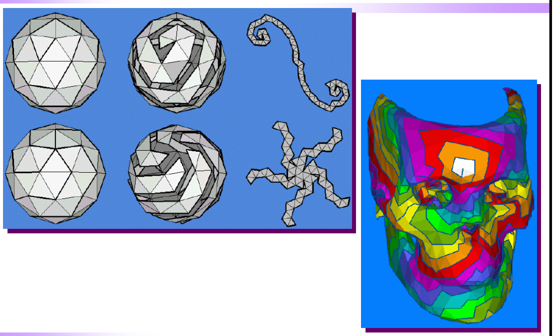
- Often more expensive than geometry.
- Exploit coherence
  - 1D
    - Vertex/Triangle tree.
  - 2D
    - 2D adjacency of faces and vertices.

Connectivity compression [Taubin98a]

- Taubin's "topological surgery" method [Taubin98a]
  - Cut out a manifold mesh to create
    - Spanning tree of vertices and triangle strips (trees).
    - Efficiently encode the above two.



Connectivity compression [Taubin98a]



2000/06/23 Yamanashi University Ryutarou Ohbuchi 25

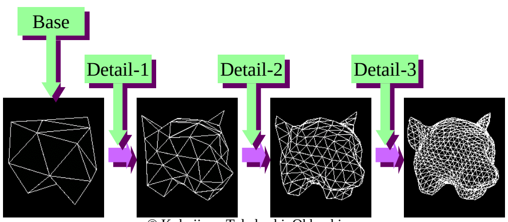
Connectivity compression [Taubin98a]

- Process
  - Find spanning trees of vertices.
  - Cut out the mesh along the spanning tree to generate triangle strips.
  - Encode triangle strips and spanning trees.
    - Best case at 2 bit/Triangle to encode triangle strips.
    - Further compression by run-length encoding.
- Efficiency: 1-2 byte/triangle
  - Connectivity: ~2.2 bits/triangle
    - Increases for smaller meshes
  - Geometry: 4-6 bits per coordinate
    - Decreases with tessellation & quantization

2000/06/23 Yamanashi University Ryutarou Ohbuchi 26

Progressive compression

- Encode and transmit simple base model
- Refine by adding incremental details.

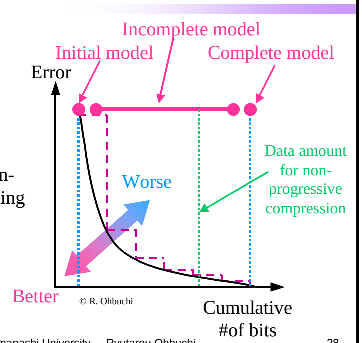


© Kokojima, Takahashi, Ohbuchi

2000/06/23 Yamanashi University Ryutarou Ohbuchi 27

Progressive compression

- Granurality
  - Single vertex
  - Entire model
- Overhead
  - Data size % of non-progressive encoding



2000/06/23 Yamanashi University Ryutarou Ohbuchi 28

Progressive compression [Cohen-Or99]

- Characteristics
  - Multiresolution analysis-like approach.
  - Small granularity (e.g., a vertex).
  - Relatively large initial model (i.e., coarsest approximation)
- Method
  - Vertex removal and re-triangulation
    - Encode triangles involved in the vertex removal by triangle "coloring".
    - The coloring method method misses some exceptional triangle configurations, which are left in the base model.
  - 2D vertex coordinate prediction
    - Predict from adjacent vertices.

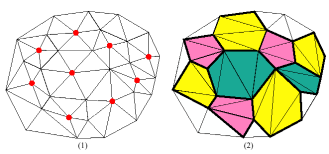
2000/06/23 Yamanashi University Ryutarou Ohbuchi 29

Progressive compression [Cohen-Or99]

- 4-color encoding
  - 3 colors are not always enough to describe arbitrary topology.
    - Simply leave such cases in the "base" mesh.
  - Efficiency at 2 bit/triangle
    - Per-vertex figure depends on a vertex's valence.

(1) Red vertices are to be removed.

(2) Polygons are re-tiled. There are 4 cases for the re-tiling (including null case).



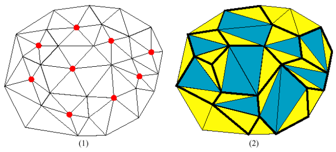
2000/06/23 Yamanashi University Ryutarou Ohbuchi 30

Progressive compression [Cohen-Or99]

- 2-color encoding
  - Further restriction on triangulation.
  - Efficiency at 1bit/triangle.
    - Per-vertex figure depends on the vertex's valence.
    - Ex. Removal of a 6-degree vertex requires 4 bits, and a 5-degree vertex requires 3bits.

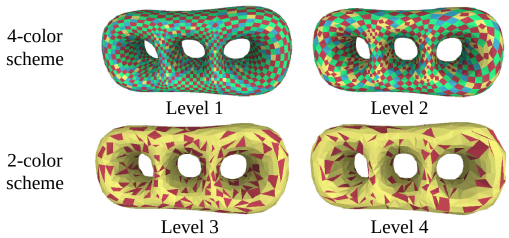
(1) Red vertices are to be removed.

(2) Polygons are re-tiled. There are only 2 cases for the re-tiling (including null case).



Progressive compression [Cohen-Or99]

- Example of coloring
  - 4-and 2-color schemes used on different (resolution level) meshes.



Progressive compression [Cohen-Or99]

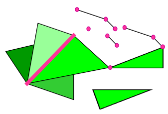
- Close to the best (so far) non-progressive compression scheme by Touma & Gotsman (T&G) [Touma98].
  - ~5.98bit/vertex

	Cohen-Or (progressive)			T&G	Ratio
model	stream	base	total	TG	ratio
horse	42001	10445	52446	47108	1.11
jaw	29533	8090	37623	34577	1.08
blob	19928	5757	25685	21396	1.18
hole3	8182	4001	12183	13452	0.90
tricerotops	7252	2026	9278	7871	1.17

12bit/coordinate quantization, sizes in bytes.

Nonmanifolds

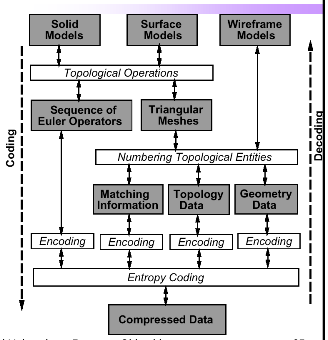
- Dirty data
  - Typical VRML models are "dirty"
    - Full of non-manifold, isolated points/isolated polygons.
  - Analytical surfaces may be non-orientable.
  - CAD data may contain non-manifold.features.



- Efficient compression requires manifold adjacency.
- Compressing non-manifold models
  - Convert into pseudo-manifold and compress.
    - Connectivity lost.
  - Convert into pseudo-manifold, save conversion operation.
    - Original model can be recovered.

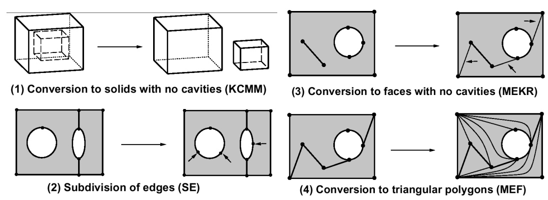
Nonmanifolds [Masuda98]

- Convert topology into manifolds
  - To triangular mesh.
  - Record operator sequence.
  - Relate original model and its attributes.
- Encode triangular mesh.
- Encode operator sequence.



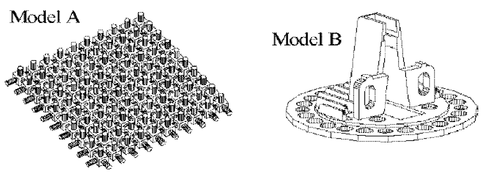
Nonmanifolds [Masuda98]

- Topology conversion operators
  - Small operation size (4) ensures very efficient encoding.



Nonmanifolds [Masuda98]

- Compression results



15,600edges, 5,520faces   3,304edges, 1,116 faces

Nonmanifolds [Masuda98]

- Compression results

	Models	Original (100%)	gzip	Our method
A	WF topology only	153.89	54.75 (35.58%)	0.23 (0.15%)
	WF w/geometry	403.49	87.86 (21.78%)	3.66 (0.91%)
	Solid topology only	495.52	186.59 (37.66%)	11.43 (0.31%)
	Solid w/geometry	745.12	219.70 (29.49%)	11.43 (0.41%)
B	WF topology only	29.72	12.67 (42.63%)	0.66 (2.22%)
	WF w/geometry	82.48	23.81 (28.87%)	6.73 (8.16%)
	Solid topology	93.35	38.80 (41.56%)	4.21 (4.51%)
	Solid w/geometry	146.11	49.94 (34.18%)	10.28 (7.03%)

Time dependent geometry & connectivity

- Geometry change
  - Rigid body: 6 DOF parameter stream.
    - MPEG-4 has compression scheme for this.
  - Soft body: Vertex coordinate stream.
- Connectivity change
  - MPEG-4 Delaunay mesh (2D).
- Everything change
  - HoloFlick™ [Deering99]
    - Requires full-on streaming.

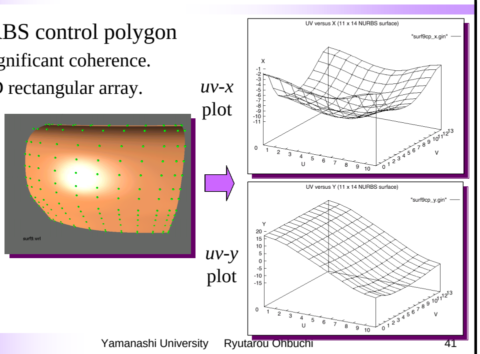
Research opportunities here!

Frequency domain techniques

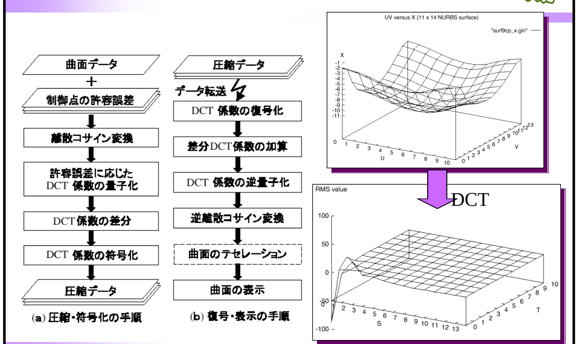
- Lossy compression of NURBS surfaces [Masuda99]
  - Lossy compression acceptable for certain CAD applications.
    - E.g., parts or design preview.
  - Issue: "Cracks" between patches must be dealt with [Mukai99]
- Lossy compression of polygonal meshes [Karni00]
  - Eigenvalue of the eigenvector computed from a mesh "Laplacian" as "frequency" coefficients.
  - Remove insignificant frequency component for compression.

Lossy NURBS compression [Masuda99]

- NURBS control polygon
  - Significant coherence.
  - 2D rectangular array.



Lossy NURBS compression [Masuda99]



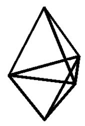
Frequency domain polygon compression [Karni00]

- Lossy, but smooth approximation with few coefficients.
- Laplacian L of a mesh???
  - Aij : adjacency matrix
  - ***D*** : Diagonal matrix s.t. $D_{ij}=1/d_{ij}$ where d_{ij} is the valence of vertex i.
  - Laplacian L of a mesh is defined as $\boldsymbol{L}=\boldsymbol{I}-\boldsymbol{DA}$ where

$$L_{ij} = \begin{cases} 1 & i = j \\ -1/d_{ij} & i \text{ and } j \text{ are neighbours.} \\ 0 & \text{Otherwise} \end{cases}$$

- Eigenvectors of L form an orthogonal basis of R^n.
  - Associated eigenvalues may be considered as ***frequencies.***

Frequency domain polygon compression [Karni00]



model

$$\begin{bmatrix} 0&1&1&1&1 \\ 1&0&1&1&1 \\ 1&1&0&1&1 \\ 1&1&1&0&0 \\ 1&1&1&0&0 \end{bmatrix} \quad \begin{bmatrix} 1.00&-0.25&-0.25&-0.25&-0.25 \\ -0.25&1.00&-0.25&-0.25&-0.25 \\ -0.25&-0.25&1.00&-0.25&-0.25 \\ -0.33&-0.33&-0.33&1.00&0 \\ -0.33&-0.33&-0.33&0&1.00 \end{bmatrix}$$

Adjacency matrix — Laplacian

$$\begin{bmatrix} -0.447&0.000&-0.817&0.000&0.366 \\ -0.447&0.000&0.408&-0.707&0.366 \\ -0.447&0.000&0.408&0.707&0.366 \\ -0.447&0.707&0.000&0.000&-0.548 \\ -0.447&-0.707&0.000&0.000&-0.548 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 & 1.25 & 1.25 & 1.51 \end{bmatrix}$$

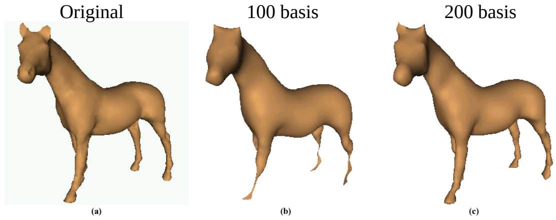
Eigenvector — Eigenvalue

Eigenvalue of DC (constant) engenvector is 0.

Frequency domain polygon compression [Karni00]

- Example
  - Original horse model analyzed by 2978 basis functions

Original 100 basis 200 basis

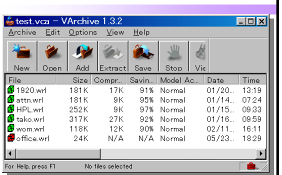


Products and standards

- MPEG4
  - [Taubin98a] algorithm.
    - Static geometry & face- and body-animation.
- Java3D
  - [Deering95] algorithm.
- Virtue, Ltd. “VArchive” (www.virtue.com)
  - [Touma98] algorithm.
  - Commercial 3D model archival tool.
- MetaCreations/Intel “Metastream”
  - [Abadjev99].
  - Media file format that covers geometry as well as texture image and texture information.

VArchive 3D model compression tool

- WinZip-like archival tool.
  - Compresses and archives 3D models.
  - Trial version available from
    - http://www.virtue.com



Virtue社の3次元モデル圧縮ツールVArchive

WRL files	Original [KB]	VArchive Comp'd [KB]	VArchive Saving [%]	WinZip Comp'd [KB]	WinZip Saving [%]
1920	181	17	91	29	84
Attan	181	9	95	40	78
HPL	252	9	96	53	79
Tako	317	27	91	63	80
Wom	118	12	90	26	78

VArchiveと汎用圧縮ツールWinZipとの比較

Digital Watermarking

Outline
- 3D model as multimedia datatype
- Compression
- **Digital watermarking**

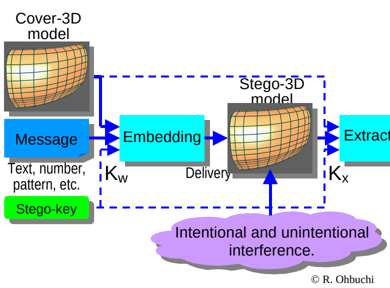
Digital Watermarking?

- Embed messages into data objects as structures called watermark.
  - Transparently to the intended applications of the objects.
  - Watermarks inseparable from the objects.

Digital watermarking

- Applications
  - Theft deterrence.
  - Copyright notification.
  - Tamper detection.
  - Content integrity check.
  - Fingerprinting
  - Annotation.
  - Delivery control (e.g., playback control)
  - Covert communication.
  - …

Digital watermarking



Watermarking Taxonomy

- Transparent v.s. non-transparent
  - Transparent: Unnoticeable without special processing.
  - Non-transparent: Noticeable but not interfering with the intended applications of the data object.
- Robust v.s. fragile
  - Robust: Survives intentional & unintentional interferences.
  - Fragile: Detects tampering by breaking down.
- Blind detection v.s. non-blind detection
  - Blind detection: Extraction doesn't require original cover-data.
  - Non-blind detection: Original cover-data used for extraction.

Digital watermarking target data types

- Image
  - [Tanaka, Bender, Cox, Braudway, O'Ruandaidh, Tewfik, Zhao, ...]
- Movie
  - [Hartung, Morimoto, ...]
- Audio
  - [Matsui, Gruhl, ...]
- Text, program
  - [Maxemchuk, …]
- 3D models
  - Static
    - [Ohbuchi97-98, Kanai98, Benedens99a, Benedens99b, Kanai99, Yeo99, Ohbuchi99, Praun99, Shiba99]
  - Dynamic
    - [Hartung98]

3D Model Watermarking

- Shape of polygonal meshes
  - Vertex coordinate modification
    - Ohbuchi97, Ohbuchi98a, Ohbuchi98b, Kanai98, Benedens99a, Benedens99b, Date99, Yeo99, Praun99.
  - Vertex topology modification
    - Ohbuchi97, Ohbuchi98a, Ohbuchi98b
- Shape of parametric curves and surfaces
  - Reparameterization, knots insertion
    - Ohbuchi99, Ohbuchi00
- Attributes of polygonal meshes
  - Per-vertex texture coordinate modification
    - Ohbuchi98b
- Others
  - MPEG4 face animation parameter
    - Hartung98

Watermarking 3D shapes

- Shape
  - Polygonal mesh
  - NURNS surfaces
- Shape attributes
- Animation

Embedding in Tri-mesh Coordiantes [Ohbuchi97]

- First of a kind.
- Modifies *local* geometrical properties of a polygonal mesh.
  - Pros
    - Robust against a class of geometrical transformation (e.g., affine transformation).
  - Cons
    - Not robust against coordinate noise or global modifications.

Shape Modification Primitives

- Geometrical primitives
  - Modify geometrical transformation *invariants*.
    - Coordinate transformation destroys coordinate values!
    - Indirectly modify vertex coordinates.
  - Ex. Affine transformation invariants
    - Ratio of lengths of line segments on a line.
    - Ratio of volumes of two polyhedrons.

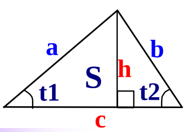
Primitive Arrangement

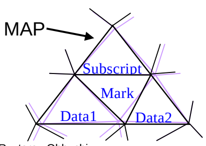
- Embed data into an arrangement of primitives.

  {g,a,r,m,t,n,r,e,e,n,a} ➡ {a-r-r-a-n-g-e-m-e-n-t}
  - Ex. Embed a visible pattern in a 2D arrangement.
  - Ex. Embed a symbol string in a 1D arrangement.
- Arrangement = (Initial condition) + (Ordering)
  - Both must withstand expected disturbances.
- Arrangement is easier in audio, image, text, or movie data.

Triangle Similarity Quadruple Method
Embed in Vertices of Poly-Meshes

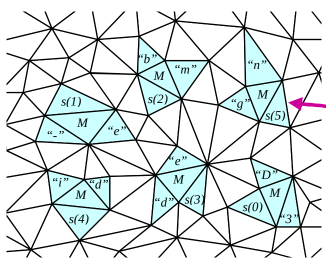
- Primitive
  - Dimension-less quantities {a/b, h/c} that defines a set of similar triangles.
  - Invariant to translation, uniform-scaling, and rotation.
- Arrangement
  - by subscript in *Macro Embedding Primitives (MAP)*





Triangle Similarity Quadruple Method
Embed in Vertices of Poly-Meshes

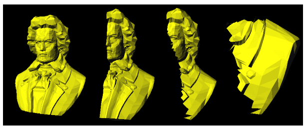
- Embedding example.



Triangle Similarity Quadruple Method
Embed in Vertices of Poly-Meshes

- Resistant to resection due to
  - Subscript-arrangement.
  - Repeated embedding.

Resection and data loss



(a) (b) (c) (d)

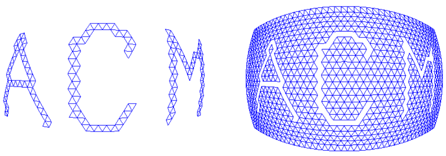
	No. of Δs	Data remained intact
a	4889	6 copies, 132 bytes each.
b	2443	132/132 bytes
c	1192	102/132 bytes
d	399	85/132 bytes

Embedding in Tri-mesh Topology [Ohbuchi97]

- Modify topological properties to embed data.
  - Encode data in how things are connected.
    - Connectivity of points, triangles, tetrahedrons, etc.
  - Unaffected by geometrical transformation.
    - Ex. Cut-out stencil patterns from a mesh.
- Topology non-existent in traditional image/audio data.
  - Topology found in polygonal meshes, irregular grid volume data, etc.

Simple Mesh Stenciling by Data Pattern

- Cut-out, or modify connectivity of a mesh.



Transformed domain approaches

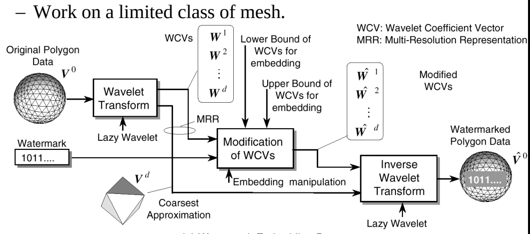
- A transformed domain approach
  - Transform a given mesh into a set of coefficients.
  - Modify coefficients according to message.
  - Reflect changes in coefficients back to the mesh.
- Pros and cons
  - More robust than non-transformed domain approaches.
  - Mesh need to be fairly complex.
  - Typically requires remeshing.

Transformed domain watermarking

- Kanai98
  - Triangular wavelet decomposition.
  - Robust against affine transformation.
  - Robust against random noise.
- Praun99
  - Progressive mesh decomposition + mesh warping by using various basis functions.
  - Robust against similarity transformation.
  - Robust against addition of random noise, polygon reduction and cropping.

Kanai's MRA watarmarking

- Embedding
  - Work on a limited class of mesh.

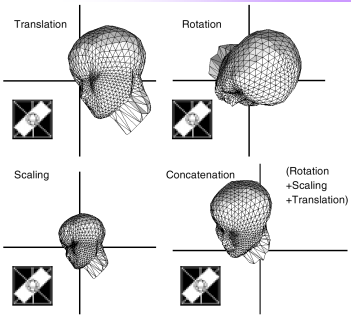


(a) Watermark Embedding Process

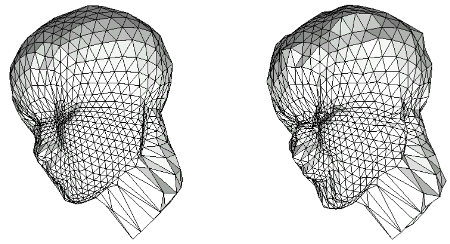
Kanai 98 (IFIP98)

Kanai's MRA watarmarking

- Robust against
  - Afine transformation
  - Addtion of random noise.



Kanai's MRA watarmarking

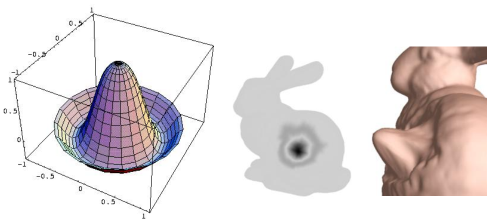


(a) Original Polygon (-10< *x,y,z*<10)

(b) Watermarked Polygon (ε =1)

Praun's watermarking

- Basis function to modify shape at various scale.
  - Modify visually significant part of shape.
- Size of the basis carries information.



Praun's watermarking

- Embed
  - Progressive-mesh based MRA to create a remeshed mesh.
  - Modify the mesh by using a scalar basis function modulated by message.

$$\begin{bmatrix} V'_x \end{bmatrix} = \begin{bmatrix} V_x \end{bmatrix} + e * \begin{bmatrix} \Phi \end{bmatrix} \begin{bmatrix} h_{1d} 1_x & & 0 \\ & \ddots & \\ 0 & & h_m d_{mx} \end{bmatrix} \begin{bmatrix} w \end{bmatrix}$$

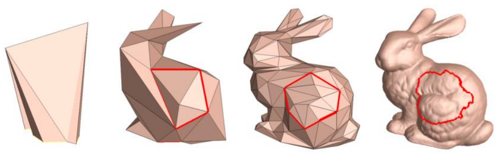
V'$_x$:modified vertices. V$_x$: original vertices.
w: message. ε: modification scale
Φ: scalar basis function, hd$_x$: displacement direction.

Praun's watermarking

- Extract
  - Align cover-mesh and stego-mesh.
    - For rotation, uniform scaling, and translation.
    - Possible human intervention necessary for alignment.
  - Resample stego-mesh to create a mesh with the topology of the original cover-mesh.
    - Tolerate polygon simplification and/or remeshing by saboteur.
  - Extract by using MRA.

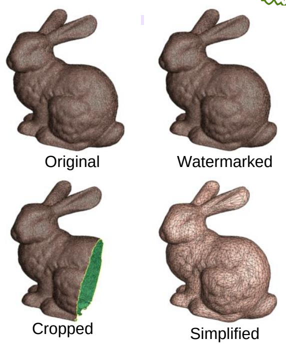
Praun's watermarking

- Multi-resolution analysis based on Hoppe's progressive mesh representation.



Praun's watermarking

- Robust against
  - Similarity transformation
  - Cropping
  - Addition of noise
  - Simplification



Praun's watarmarking

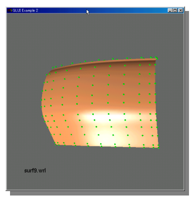
- Pros:
  - Quite robust!
    - Similarity transformation, cropping, random noise addition, and simplification.
- Cons:
  - Private watermarking.
  - Works only on fairly complex meshes.
  - Requires remeshing.
  - Slow.

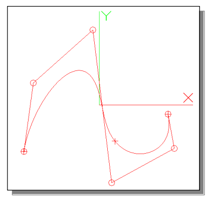
Watermarking 3D models

- Shape
  - Polygonal mesh
  - NURNS surfaces
- Shape attributes
- Animation

3D geometric CAD models?

- Parametric curves and surfaces.
- Exact shape preservation.
- Functional transparency a must.





Functionally-transparent watermarking for 3D geometric CAD models

- Innate redundancy
  - Limited shape changes won't affect functional transparency.
    - Shape altering, but functionally transparent.
    - E.g., Automobile chassis with "relief" watermark.
- Representation redundancy
  - An arbitrary number of representations exist for a same shape.
    - Shape preserving and functionally transparent.
    - E.g., A NURBS surface before and after knots insertion.
- Encoding redundancy
  - Minute changes won't affect shape.
    - Almost shape-preserving and functionally transparent.

Watermarking parametric curves and surfaces

1. Preserves shape *and* data size.
   - Reparameterization by using rational-linear functions.
2. Preserves shape, alters data-size.
   - Degree elevation
   - Knots insertion.
   - Reparameterization by using a polynomial with degree >1.

Watermarking parametric curves and surfaces

3. Alters shape, preserves data size.
   - Modify knot vector values.
   - Modify control point coordinates.
4. Others

NURBS curve representation

- NURBS curve

$$\mathbf{C}(u)=\sum_{i=0}^{n}N_{i,p}(u)w_i\mathbf{P}_i \Big/ \sum_{i=0}^{n}N_{i,p}(u)w_i$$

$\mathbf{P}_i$: control points, w_i: weights, $u\in[a, b]$

- Basis

$$N_{i,0}(u)=\begin{cases}1 & \text{if } u_i \le u < u_{i+1}\\ 0 & \text{otherwise}\end{cases}$$

$$N_{i,p}(u)=\frac{u-u_i}{u_{i+p}-u_i}N_{i,p-1}(u)+\frac{u_{i+p+1}-u}{u_{i+p+1}-u_{i+1}}N_{i+1,p-1}(u).$$

- Knot vector

$$U=\{\underbrace{a,\dots,a}_{p+1},u_{p+1},\dots,u_{m-p-1},\underbrace{b,\dots,b}_{p+1}\}$$

Watermarking by rational-linear reparameterization

- Shape preservation
  - Exact geometric shape preserved.
    - Except for a few cases, e.g., ruled surface, in which shape depends on parameterization.
- Non-blind detection
  - Extraction needs comparison with the original (non-watermarked) data.
- Fragile watermarking
  - Useful for tamper detection.
- Data-size preservation

Watermarking by rational-linear reparameterization

- NURBS curve $\mathbf{C}(u)$ (Reparameterization studied by [Lee & Lucian 91])

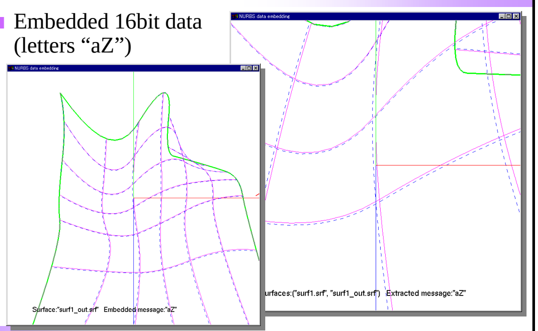
Reparameterize by $u = f(s)$

$$u = f(s)=\frac{-\delta s+\beta}{\gamma s-\alpha} \quad s\in[c,d]$$

- NURBS curve $\mathbf{C}(s)$ reparameterized by $f(s)$ has
  - the same *geometric shape*
    - A few exceptions include ruled surface.
  - the same *data size*
    - i.e., the same # of control points and knots.

Watermarking by rational-linear reparameterization

- Embedded 16bit data (letters "aZ")



Watermarking by knots insertion

- Shape preservation
  - Exact geometric shape preserved.
- Non-blind detection
  - Extraction needs comparison with the original (non-watermarked) data.
- Robust watermarking
  - This watermark is robust if intended applications require shape preservation.
  - Useful for copyright protection, fingerprinting, etc.
- Data-size increasing
  - Number of knots increases.

Watermarking by knots insertion

- Insert a new knot, whose value encodes information.
  - Other knots remain unchanged.
  - Coordinates of control points change.
- Extends trivially to NURBS surfaces.
- A new knot $\bar{u}$ may take any value in the range $u_0 < \bar{u} < u_m$

$$U = \{u_0, \ldots, u_k, \quad u_{k+1}, \ldots, u_m\}$$

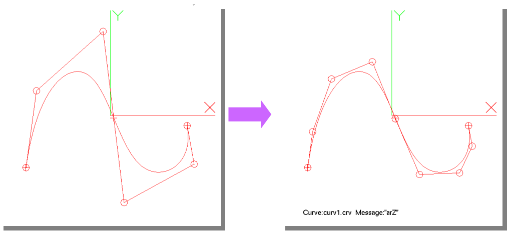
$$\downarrow \quad \downarrow \quad \downarrow \quad \downarrow$$

$$\bar{U} = \{\bar{u}_0, \ldots, \bar{u}_k, \bar{u}, \bar{u}_{k+2}, \ldots, \bar{u}_{m+1}\}$$

2000/06/23 Yamanashi University Ryutarou Ohbuchi 85

Watermarking by knots insertion

- Three knots inserted as watermark.
  - Each knot encodes 7bits of information.



2000/06/23 Yamanashi University Ryutarou Ohbuchi 86

Watermarking 3D models

- Shape
  - Polygonal mesh
  - NURNS surfaces
- Shape attributes
- Animation

2000/06/23 Yamanashi University Ryutarou Ohbuchi 87

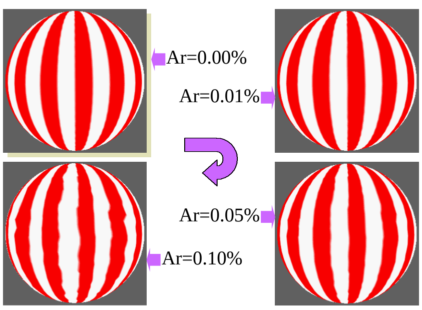
Texture Coordinate Modulation Method
[Ohbuchi98]

- Symbol sequence embedding
  - By modifying texture coordinate.
- Primitive
  - Displaced texture coordinate.
    - Texture coordinate is unaffected by geometrical transformation of vertex coordinate.
- Arrangement
  - By any one of a number of methods.
    - Ex. As they appear in the file, by geometrical quantity (e.g., area of triangles), by topology.

2000/06/23 Yamanashi University Ryutarou Ohbuchi 88

Texture Coordinate Modulation Method
A sphere (961 vertices) with simple stripe texture.

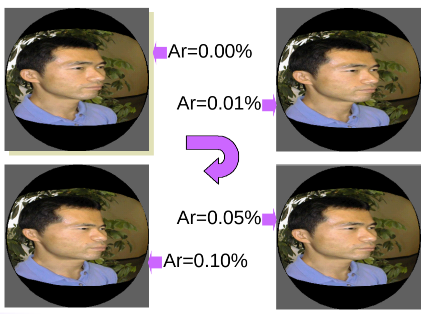
Distortion noticeable with a geometric texture.



2000/06/23 Yamanashi University Ryutarou Ohbuchi 89

Texture Coordinate Modulation Method
A sphere (961 vertices) with human face texture.

Distortion less noticeable with a complex texture.



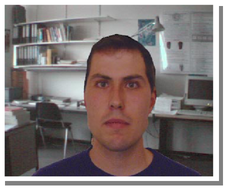
2000/06/23 Yamanashi University Ryutarou Ohbuchi 90

Watermarking 3D models

- Shape
  - Polygonal mesh
  - NURNS surfaces
- Shape attributes
- Animation

Embedding in Face Animation Parameter [Hartung98]

- Target: MPEG4 Face Animation Parameter (stream)
  - Parameters (small integers) to move components of mesh-based face models.



Embedding in Face Animation Parameter [Hartung98]

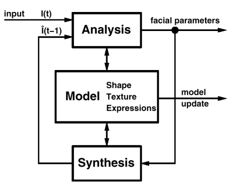
- MPEG4 Face Animation Parameters (FAPs)
  - Parts of "standard mesh" are moved by parameters, not by vertex coordinate specifications.
  - 66 parameters control a face.
    - E.g., 8 FAPs for eyebrows, 4FAPs controlling ears, etc.
  - Each parameter is typically encoded by using 5 to 7 bits .
    - Ex. "Move center of upper rip to position 17 (out of the range [0..31])"
- Face mesh is sent first. Animation parameters are sent later through streaming transmission.
  - Non-streaming mode also supported.

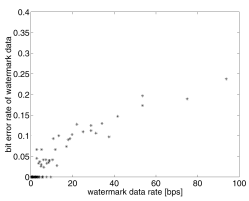
Embedding in Face Animation Parameter [Hartung98]

- Embedding
  - Modulate phase component of the FAPs
    - Spread modification over multiple FAPs.
  - Spread-spectrum communication-like "randomized dispersion" of modulation over multiple parameter streams.

Embedding in Face Animation Parameter [Hartung98]

- Extraction
  - Succeeded in extracting watermark from rendered MPEG2 (~600Kbps) video streams.
  - By using vision-based face motion recognizer/ encoder.





Summary and conclusion

- 3D Geometric CAD data are multimedia data.
  - Traditional operations
    - Input, output, and editing.
  - "Internet-oriented" operations
    - Compression, hyperlinking, indexing, similarity-based search, security-related operations, …
- Data compression for 3D model
  - Batch v.s. progressive v.s. streaming
  - Lossy v.s. loss-less
  - Topology-preservation trade-off

Summary and conclusion

- Security oriented operation using digital watermark
  - Copyright protection, authentication, tamper detection, …
- Digital watermarking
  - Add messages to media contents by adding invisible structure.
  - Useful in managing media data, e.g., for tamper detection and copyright protection.
- Bibliography listing for 3D model watermarking.
  www.kki.yamanashi.ac.jp/~ohbuchi/research/3dwmbib.html

References

- [Abadjev99] V. Abadjev, M. del Rosario, A. Lebedev, A. Migdal, and V. Pashaver, MetaStream, Proceedings of the *VRML 99*, pp. 53-62, 1999.（http://www.metacreations.com）
- [Bajaj99] Chandrajit L. Bajaj, Valerio Pascucci, Guozhong Zhuang, Progressive Compression and Transmission of Arbitrary Triangular Meshes, *Proceedings of the IEEE Visualization '99*, pp.307-316, 1999.
- [Benedens99a] O. Benedens, Geometry-Based Watermarking of 3D Models, *IEEE CG&A*, pp. 46-55, January/February 1999.
- [Benedens99b] Oliver Benedens, Two High Capacity Methods for Embedding Public Watermarks into 3D Polygonal Moels, ACM Multimedia 99 Workshop on Information Security.
- [Cohen-Or99] Daniel Cohen-Or, David Levin, Offir Remez, Progressive Compression of Arbitrary Triangular Meshes, *Proceedings of the IEEE Visualization '99*, pp.67-72, 1999.

References

- [Deering95] M. Deering, Geometric Compression, *Proceedings of the ACM SIGGRAPH '95*, pp. 13-20, 1995.
- [Gumhold98] S. Gumhold, and W. Strasser, Real Time Compression of Triangle Mesh Connectivity, *Proc. ACM SIGGRAPH '98*, Orlando, Florida, USA, 1998.
- [Gumhold99] Stefan Gumhold, Stefan Guthe, Wolfgang Straßer, Tetrahedral Mesh Compression with the Cut-Border Machine, *Proceedings of the IEEE Visualization '99*, pp.51-58, 1999.
- [Guéziec99] André Guéziec, Frank Bossen, Gabriel Taubin, Claudio Silva, Efficient Compression of Non-Manifold Polygonal Meshes, *Proceedings of the IEEE Visualization '99*, pp.73-80, 1999.
- [Hartung98] F. Hartung, P. Eisert, and B. Girod, Digital Watermarking of MPEG-4 Facial Animation Parameters, *Computer and Graphics*, Vol. 22, No. 4, pp. 425-435, Elsevier, 1998.

References

- [Hoppe96] H. Hoppe, Progressive Meshes, Proc. *ACM SIGGRAPH '96*, pp. 99-108, 1996.
- [ISO-VRML] ISO/IEC 14772-1 Virtual Reality Model Language (VRML).
- [ISO-MPEG4] ISO/IEC JTC1/SC29/WG11 *MPEG-4 Visual* and *MPEG 4 SNHC*.
- [Kanai98] S. Kanai, H. Date, and T. Kishinami, Digital Watermarking for 3D Polygons using Multiresolution Wavelet Decomposition, Proc. of the *Sixth IFIP WG 5.2 International Workshop on Geometric Modelling: Fundamentals and Applications (GEO-6)*, pp. 296-307, Tokyo, Japan, December 1998.
- [Date99] Hiroaki Date, Satoshi kanai, Takeshi Kishinami, Digital Watermarking for 3D Polygonal Model Based on Wavelet Transform, Proceedings of DETC '99, 1999 ASME Design Engineering Technical Conferences, September 12-15, 1999, Las Vegas, Nevada.

References

- [Karni00] Zachi Karni, Craig Gotsman, Spectral Compression of Mesh Geometry, to be published in the proceedings of the SIGGRAPH 2000 (http://www.cs.technion.ac.il/~gotsman/publication.html)
- [Koh99] Bengliang Koh, Tsuhan Chen, Progressive Browsing of 3D Models, *IEEE Signal Processing Society 1999 Workshop on Multimedia Signal Processing* (Electronic proceedings), 1999. (also at http://amp.ece.cmu.edu/Publication/Bengliang/mmsp/index.htm)
- [Li98] J. Li, and C.-C. J. Kuo, Progressive Coding of 3-D Graphic Models, *Proceedings of the IEEE, Special Issue on Multimedia Signal Processing*, Vol. 86, No. 6, pp. 1052-1063, June 1998
- [Masuda98] **増田 宏，大渕 竜太郎，青野 雅樹，位相操作を用いた3次元形状モデルのデータ圧縮法，情報処理学会論文誌，第39巻 第7号，** pp. 2189-2195, 1998**年7月号.**
- [Masuds99] **増田 宏，大渕 竜太郎，青野 雅樹，周波数領域での曲面データの圧縮と転送，情報処理学会論文誌，第40巻 第3号，** pp. 1188-1195, 1999**年3月号.**

References

- [Menezes96] A. J. Menezes, P. C. van Oorshot, and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1996.
- [Ohbuchi97] R. Ohbuchi, H. Masuda, and M. Aono, Watermarking Three-Dimensional Polygonal Models, *Proceedings of the ACM Multimedia '97*, Seattle, Washington, USA, November 1997, pp. 261-272.
- [Ohbuchi98a] R. Ohbuchi, H. Masuda, and M. Aono, Watermarking Three-Dimensional Polygonal Models Through Geometric and Topological Modifications, pp. 551-560, *IEEE Journal on Selected Areas in Communications*, May 1998.
- [Ohbuchi98b] R. Ohbuchi, H. Masuda, and M. Aono, Geometrical and Non-geometrical Targets for Data Embedding in Three-Dimensional Polygonal Models, *Computer Communications*, Vol. 21, pp. 1344-1354, Elsevier (1998).
- [Ohbuchi99] R. Ohbuchi, H. Masuda, and M. Aono, A Shape-Preserving Data Embedding Algorithm for NURBS Curves and Surfaces, In Proceedings of *Computer Graphics International '99 (CGI '99)*, pp. 180-187, Canmore, Canada, June 4-June 11, 1999.

References

- [Pajarola99] Renato Pajarola, Jarek Rossignac, Andrzej Szymczak, Implant Sprays: Compression of Progressive Tetrahedral Mesh Connectivity, *Proceedings of the IEEE Visualization '99*, pp.299-305, 1999.
- [Pajarola00] Renato Pajarola, Jarek Rossignac, Compressed Progressive Meshes, *IEEE Trans. Visualization and Computer Graphics*, Vol. 6, No. 1, January-March, 2000.
- [Piegl97] L. Piegl, W. Tiller, *The NURBS Book*, 2nd Edition, Springer, Berlin, 1997.
- [Popociv97] J. Popovic and H. Hoppe, Progressive Simplicial Complexes, Proceedings of the *ACM SIGGRAPH '97*, pp. 217-224, 1997.
- [Rossignac99a] J. Rossignac, Edgebreaker: Connectivity compression for triangle meshes, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 5, No.1, January-March 1999.

References

- [Rossginac99b] J. Rossignac and G. Taubin, organizers, 3D Geometric Compression, *ACM SIGGRAPH '99 Course Notes #22*, 1999.
- [Szymczak99] A. Szymczak, J. Rossignac, Grow & Fold: Compression of tetrahedral meshes, Graphics and Visualization Center Technical Report GVU-99-02, Georgia Institute of Technology, (to appear in Solid Modeling 99).
- [Taubin98a] G. Taubin and J. Rossignac, Geometry Compression through Topological Surgery, *ACM Transactions on Graphics*, Vol. 17, No. 2, April 1998.
- [Taubin98b] G. Taubin, A. Guéziec, W.P. Horn, and F. Lazarus, Progressive Forest Split Compression, *Proc. ACM SIGGRAPH '98*, pp. 123-132, 1998.
- [Touma99] C. Touma, and C. Gotsman, Triangle Mesh Compression, *Proceedings of Graphics Interface '98*, Vancouver, BC, June 1998.
- [Yeo99] B-L. Yeo and M. M. Yeung, Watermarking 3D Objects for Verification, *IEEE CG&A*, pp. 36-45, January/February 1999.