

メディアデータ型としての3次元モデル

Three-dimensional model as a media data type

大淵 竜太郎 (山梨大学) ohbuchi@acm.org

Ryutarou OHBUCHI, Yamanashi University, 4-3-11 Takeda, Kofu-shi, Yamanashi-ken, 400-8511, Japan.

Three-dimensional (3D) model has recently been recognized as one of “multimedia” data types, along with such data types as sound, text, still image, and movie. Each one of these data types has its own set of operations, which include, but not limited to, input, output, indexing, hyperlinking, compression, similarity-based search, and some form of intellectual property (IP) protection mechanism.

This paper examines a set of operations necessary for 3D models that are used for CAD and other applications. In particular, the paper concentrates on *digital watermarking* as a possible method for IP protection and *3D geometric data compression* which facilitates network based sharing and distribution of 3D models.

Key Words: Three-dimensional geometric model, data compression, digital watermarking, CAD.

1. 始めに

音, 文字, 静止画像, 動画像などのメディアデータ型は, いわゆる「マルチメディア」コンテンツの構成要素として広く使われてきた。ここ数年, これらのデータ型に加え, 3次元モデルもひとつのメディアデータ型として認知され, 他のデータ型と組み合わせて使用される機会が増えつつある。これは, PCの性能向上と, インターネットにおけるVRMLの普及によるところが大きい。

機械系の3次元CADモデルも立派なマルチメディアデータ型のひとつである。これは, 設計のドキュメント化, 協調デザイン, コンカレントデザイン, 設計の再利用, Web経由の協業・営業, 保守支援などの局面を考えるとよくわかる。

マルチメディアのメディアデータ型は, これまでテキスト, 静止画像, 音, 動画像が主であった。これらを抽象データ型として見たとき, それぞれに対応する操作の集合が用意されている。これらの操作には, 例えば, 入力, 出力, 編集, 圧縮, 索引付け, ハイパーリンク付加, 内容による検索, さらに暗号化や電子透かし埋め込み等が考えられる。これらの操作はテキストに対してはかなりよく理解されており, 著作権のような社会的・法的インフラストラクチャを要する仕組みも実際に運用されている。

しかし, テキスト以外のデータ型においてこれらの操作がみなきちんと理解され用意されているとはいえない。例えば静止画像の場合, 圧縮や入出力, 編集は実用になっているが, その内容による検索は研究から実用に踏み出したばかりだ。音や動画像のような時間変化するデータはハイパーリンクや検索, 編集などにおいて困難が生じる。例えば, 「この動画の何時の何(領域や物体)から何時の何へ」のようにリンクを指定するのは容易ではないし, 「こんな形がこんな風に動くシーン」の検索も困難な問題を含んでいる。

次節では, まず, メディアデータ型として見たときに3次元モデルに対して要求される種々の操作を概観する。ついで, その中で特に電子透かしと圧縮の技術について, その研究と実用化の現状を紹介する。

2. メディアデータ型としての3次元モデル

まず始めに, 画像と同様に, 3次元モデルにも時間に依存する動的なものとして静的なものとの2種類が存在することを注意しておきたい。動的3次元形状データにもいくつか種類があり, MPEG4のFace Animationのように幾何形状を持つ部品が位置パラメタの時間系列で動くものから, VRMLに見られるようにメッシュの頂点座標を変化させてメッシュの移動や変形を行うもの, さらにMichael Deeringの提唱するHoloFlick [Rossignac99b]のように映画の駒のような速度(例えば30コマ/秒)で動的に全ての形状(位相と座標値)を入れ替えるものなどがある。また, 形状だけでなく色や透明度などの形状に付随する属性も動的, 静的な

ものに分類できる。それが音であろうと3次元モデルであろうと, 動的なメディアの内容による類似検索やリンク付けなどの操作は困難だ。本論分では3次元モデルの時間依存性について詳しくは触れないが, 今後動的なモデルがますます重要になることは確かだ。このあたりにはたくさん研究のテーマがある。

さて, 3次元形状モデルの最も重要な性質はその3次元の形である。形は, 一般に, 点, 線, 面, ソリッドなどの形状プリミティブとそれらを組み合わせたシーンの階層で定義される。形状プリミティブにはNon-Uniform Rational B-Spline (NURBS) 曲面, ポリゴン, ボクセルなど各種が存在する。形状には, しばしばそれに付随する属性が付けられていることが多い。頂点色, 頂点法線ベクタのような写真光学的属性に加え, 温度やヤング率等の物理的属性が付随する場合もある。

3次元モデルにはどのような操作が用意されているだろうか。テキストや画像についてはかなりよくわかっている入出力も3次元モデルに付いてはまだ発展途中である。かつて3次元形状は形状モデラ(場合によってはテキストエディタかも知れないが)によって作られたが, 最近は実物体からレーザーレンジスキャナなどで取りこむ例も増えてきた。また, 色や反射率などの写真光学的属性も, モデラで指定することも取りこむことも出来る。これら形状や属性の入力技術については, 多くのサーベイ(例えば[Fitzmaurice99])がある。出力はレンダリングを経てディスプレイで視覚的に呈示するのが最も一般的だが, Phantomなどのデバイスによる力覚表示も徐々に広まりつつあるし, 視覚と力覚を同時に刺激できる, ラピッドプロトタイプング技術による直接的な形状出力も重要だ。

インターネットの普及をきっかけに急速に重要になってきた操作が圧縮, そして知的所有権(Intellectual Property; IP)管理である。Web上のVRMLやMPEG4による「ポリゴン放送」は言うに及ばず, いわゆる仮想エンタープライズにおける3次元CADモデルの共有等の応用などを考えると, これらの操作の重要性が理解される。以下, 3節4節で圧縮と, 知的所有権管理の一手法と見ることで電子透かしについてやや詳しく見て行く。

3. 3次元モデルの圧縮

Boeing 777 ジェットのポリゴンモデルは, リベットのためのポリゴンを無視してもまだ 1×10^9 ポリゴンからなるという。これをそのまま単純に1座標あたり4バイトの単精度浮動小数点3つの表現に直すと座標値だけで12Gbyteである。後述するが, 点から辺, 辺から面, 等を作るための接続情報(位相)は座標値以上のデータ量になることが多く, これも含めると20Gbyte以上にもなる。NURBSなどの曲面からなるモデルは一般にポリゴンモデルよりは小さい。しかし, 自動車1台分の外板のCADモデルをネットワーク転送することを考えるとそのデータサイズは無視できない。

3 次元モデルに限らず大きなデータを蓄積しあるいは転送するとき圧縮を考えるのは当然である。3 次元モデルの圧縮には gzip のような汎用の圧縮アルゴリズムを用いることも出来るが、一般に、圧縮しようとするデータの性質を考慮した圧縮手法を使用すると圧縮率を高めることが出来る。

データ圧縮法は、圧縮して伸長したときに圧縮前と完全に同じデータが復元される無損失圧縮(loss-less compression)と、多少異なる近似データが復元される損失圧縮(lossy compression)の2つに分けられる。また、圧縮したものを転送する場合には、全体を1度にする一括転送と、まず概形を転送し、その後通信路のバンド幅が許す速度で徐々に詳細を転送して形状を復元して行く漸近転送(progressive transmission)の2種類が考えられる。一般に、最大の詳細度を得るために必要なデータ量は一括転送より漸近転送の方が大きい、つまり後者のほうが圧縮効率が悪い。

3 次元モデルの圧縮の研究の殆どは、現在のところポリゴンメッシュ(殆どが3 角形メッシュ)の形状およびその属性の圧縮に関するものである。ポリゴンメッシュは、

- ① メッシュの頂点座標値、
- ② メッシュ頂点の接続関係(位相)、
- ③ 頂点や辺などに付随する属性値、

の3つによって決まるが、積極的に圧縮しないとすると、これらの情報の表現にはどのくらいのデータ量を要するのであろうか。

頂点座標は、モデルを囲む最小の直方体 bounding box (拘束箱)で正規化したのち必要最小限のビット数で量子化すると仮定する。すると座標軸ごとに16bit 程度あれば十分で、頂点毎に48bit 程度を要することになる。

頂点位相の表現に通常の indexed face set を使うとすると、全ての頂点に番号を振り、その3つ組で1つの3 角形を表現する。このとき、頂点個数を V とすると、3 角形1つ当たりの位相の表現に $3 \cdot \log_2 V$ bit を要する。例えば、あるメッシュが $V = 4096$ 個の頂点からなるとすると、位相の表現に3 角形1つ当たり36bitが必要となる。

属性値は何種類も考えられる。頂点毎の色はしばしば赤緑青各色8bitで表現され、頂点当たり24bitとなる。頂点法線ベクタは浮動小数点数の3 組で表現されることが多いが、正規化されているために大きさが1であること($n_x^2 + n_y^2 + n_z^2 = 1$)より、固定小数点数の2 組で十分なはずで、全体として16bit から32bit で足りるはずである。

3 次元モデル圧縮研究の最初のブレークスルーは Michael Deering の論文[Deering94]であった。彼はこの論文で、「人がブラウザで見る」使い方を前提に、頂点座標値、頂点接続関係、および各種の属性、たとえば頂点色、頂点法線ベクタなどを損失圧縮する手法を述べた。頂点座標はそのモデルの拘束箱で正規化したのち必要最小限のビット数で量子化する。この正規化を行うと頂点座標の量子化は10-12bit の分解能でほぼ十分である。もちろん、量子化された座標値はさらに Huffman 符号を用いてエントロピー圧縮する。頂点接続関係はトライアングルストリップを多少拡張した手法を用いて圧縮した。法線ベクタも圧縮しており、法線ベクタ1つを8bit-12bit で表現してもそれほどレンダリング品質が落ちないことを実験で示している。この圧縮手法の大きな目的はハードウェアのグラフィックスアクセラレータのバスバンド幅を減らすことにあり、したがってハードウェア実装を意識している。そのアルゴリズムは簡単で、また後で述べる Taubin の手法などとは異なって伸長に要するメモリ量が小さい。Deering の技術の殆どは Java3D に取り込まれた。

圧縮における次のブレークスルーは Taubin らによるメッシュ位相の圧縮技術である[Taubin98]。この手法は、三角メッシュの全頂点を spanning tree (全域木)で覆うことから始まる。こうして出来た頂点の木と、それから導かれる三角形の木を符号化することで、

それまでの手法に比べて非常に小さなビット数で3 角メッシュの位相を表現することに成功した。圧縮しようとするモデルによって異なるが、3 角メッシュの接続関係は3 角形当たり平均2.2bit で表現できた。彼らは座標値の圧縮も行っており、座標当たり4-6bit つまり3 角形あたり12-18bit あれば大抵は十分という。この Taubin らの手法は多少の改良の後 MPEG-4 標準[MPEG4]に取りこまれた。

Taubin の手法に続いて数多くの位相情報圧縮アルゴリズムが登場した。なかでも Touma ら [Touma98]、Gumhold ら [Gumhold98]、Rossignac ら [Rossignac99a]などの手法は Taubin の手法よりも位相の圧縮率が高く、いったいどこまでビットレートが下がるのか注目される。

圧縮で忘れて成らないのは漸近転送である。漸近転送では、まず大まかな形を転送し、ついで通信路の帯域幅により許される速度で何度かに分けて徐々に詳細を追加する。誰もが静止画像の漸近転送を1 度は経験したことがあると思う。漸近転送はいわゆる LOD (Levels-Of-Detail) や単純化と関係があり、単純化の手法として作られた Hoppe のプログレッシブメッシュ [Hoppe97, Hoppe98] などは漸近転送に転用が出来る。漸近転送アルゴリズムにはこの他、Hoppeの手法の変形 [Pajarola99]、上記の Taubin の手法の変形 [Taubin98]、頂点除去によるもの [Li98] などがある。

ポリゴンメッシュからなる3次元形状の圧縮は、漸近転送もふくめ実用化に向かって動いている。MPEG-4 標準は言うまでも無く、Virtue 社 [Virtue3D] や MetaCreation 社 [Abadjev99] はそれぞれ圧縮データフォーマットやアーカイブソフトウェア、ブラウザプラグインのビューアなどを製品としている。デモ版などがダウンロードできるので試してみると良い。また、ポリゴンの圧縮については [Rossignac99b]によくまとめられている。

最後に、NURBS などの曲面の圧縮について簡単に触れる。同じ品質を仮定したとき、曲面のデータ量はそのポリゴン近似のデータ量より小さい。しかし、その曲面モデルでも1 モデルで数十から数百 Mbyte になることは珍しくない。また、CAD データといえどもその利用法によっては、絶対的な精度を要求されない場合もある。例えば目視による形状デザイン評価や干渉検査がそれである。こうした仮定の下で、[増田 99]は損失のある曲面の圧縮法を提案した。パラメトリック曲面の制御点座標値には強い相関がある。増田らの手法はこれを利用し、制御点座標値の2次元配列を単なる数値として扱って離散コサイン変換(DCT)し、そのDCT 係数を量子化、エントロピー圧縮した。これまで曲面を別の曲面(基底)の線形結合で近似するような手法は数多くあったが、制御点を単なる数値列として近似し、圧縮を行う手法は[増田99]だけと思われる。

4. 電子透かし

データ埋め込み、または電子透かしと呼ばれる技術は、watermark (透かし)と呼ばれる何らかの構造体を、その存在が埋め込み対象となるコンテンツ本来の目的(例えば人による表示・鑑賞)を阻害しないようコンテンツに付加する[松井 98]。透かしをコンテンツに追加することを(データ)埋め込み(embedding)と呼び、また埋め込まれた情報を目的に応じて取り出すことを取り出し(extraction)と呼ぶ。埋め込まれた透かしは、説明の付加、改ざんの検出、あるいは正規の購入者の認証など、そのコンテンツを何らかの形で管理する目的で用いることが出来る。

4.1 電子透かしの分類

電子透かしはその可知性(perceptibility)により分類できる。不可知な透かしは何らかの機械的な処理を経て初めて可知となる。可知な透かしはそのような処理無しに観測者が知覚できる。これまでの電子透かしはその殆どが不可知な透かしである。しかし、コンテンツの無許可の再配布を牽制する目的で、可知な透かしを積極的に使った例もある。ここで注意したいのは、3次元モデ

ルの可知性の定義はそれ以外のメディアデータ型の場合よりずっと複雑なことである。例えば静止画の電子透かしでは、観測者は人であり、観測の過程は直接的である。3次元モデルでは、観測者が人の場合も、照明、カメラパラメータ、レンダリング手法などによりその観測はより間接的である。さらに、CADモデルの透かしの観測者はConstructive Solid Geometry (CSG) アルゴリズムかも知れず、この場合、透かしの有無によってCSG演算結果が変わらぬことを求められるだろう。

電子透かしはまた頑強な透かし (robust watermark) と脆弱な透かし (fragile watermark) に分類できる。前者はコンテンツに加えられる意図的・非意図的な変更に対して保存される必要がある。脆弱な透かしは、コンテンツに対する意図的な変更によって透かしが改変または破壊されることを積極的に利用し、そのコンテンツの改ざんや変更の検出を行う。ここで非意図的な変更とは、コンテンツの通常の使用において予期されるもの、例えば静止画像におけるJPEG圧縮・伸長をさす。これに対し意図的な変更とは、透かしを変更しあるいは破壊するなどの目的で意図的に加えられるものを指す。

この他、取り出しに元の (埋め込み前の) データが必要な透かしを秘密透かし (private watermark) と呼び、そうでないものを公開透かし (public watermark) と呼ぶ。後者の方便い道が広く望ましいが、頑強性などの点で技術的にはより困難である。

ここで記憶にとどめて欲しいのは、その他の各種セキュリティ技術と同様に、電子透かし技術単体では著作権保護などの現実の応用で効果を上げることはできない点である。実際に何が 필요한かはそれぞれの応用で異なるが、透かしと組み合わせて使う電子署名や、透かしを入れたデータを預託する為の信用できる預託機関などのインフラストラクチャ、そしてこれらインフラストラクチャの社会的・法的な認知など、満たされねばならない要件は数多い。

4.2 3次元モデルの電子透かし

これまでのデータ埋め込みの研究の多くは「古典的」マルチメディアデータ型、例えば文字文書、静止画像、動画像、および音声データに対する埋め込みを中心としていた。しかし最近、3次元 (3D) データがその重要性を増すにつれ [VRML] [MPEG4], 3Dモデルに対する埋め込みアルゴリズムがいくつか発表された。以下、3次元モデルに対する透かしをその埋め込みの対象で分類し紹介する。3次元形状を表現するために最も広く用いられているのがポリゴンメッシュであり、まずポリゴンメッシュへの透かしの埋め込み法を紹介する。圧縮の節で見たように、静的なポリゴンメッシュには頂点座標、頂点接続関係、属性、の3種の構成要素があり、これら全てが電子透かし埋め込み対象となる。その後、われわれが現在研究中のパラメトリック曲線・曲面に対する透かしの埋め込み手法についてのべる。

4.2.1 ポリゴンメッシュの形状への埋め込み

座標値の変更による埋め込み：我々が提案したアルゴリズムのうち幾つかは、ポリゴンメッシュによって定義された形状の座標値を埋め込みの対象とする。座標値そのままでは、たとえば平行移動や回転で消えてしまう。そこで我々はあるクラスの幾何変換、たとえば相似変換やアフィン変換に対する不変量を利用して透かしを埋め込んだ [Ohbuchi97, Ohbuchi98a, Ohbuchi98b]。この不変量は頂点座標から誘導されたもので、アフィン変換不変量の一例としては4面体の体積の比がある。不変量を用いたことで、3Dモデルが日常的にさらされる可能性のある幾何変換、例えばアフィン変換をかけられても透かしは壊れない。ただ、この種の透かしは頂点座標へのノイズの重畳により破壊しやすい。ほかに、[Shiba99] は、アフィン変換普遍量である直線状の長さの比を操作して埋め込むが、メッシュを切りなおすためデータ量は増える。

Minerva Yeung らは2次元の静止画像を対象として、fragile watermark (脆い透かし、後述) の概念を、初めて提唱した。この

脆い透かしを同じグループが3Dモデルに適用した例が [Yeo99] に述べられている。この手法では頂点座標値を変更して脆い透かしを埋め込んでいる。

頂点座標値への変換領域での埋め込み：金井らは3角メッシュをウェーブレット変換し、その変換領域の係数を操作して情報を埋め込む手法を提案した [Kanai98]。変換領域で操作した係数を逆変換すると透かしを埋め込んだ形状が得られる。彼らの透かしはアファイン変換に耐え、かつ頂点座標に重畳されたランダムノイズに対してもある程度の頑強性を有した。Praun らの手法 [Praun99] の考え方は Kanai らの手法に近いが、メッシュを直接ウェーブレット変換するのではなく、与えられたメッシュ上に多重解像度の基底関数 (正規性や直交性は持たない) を立て、その係数を変更することで透かしの埋め込みを行った。Benedens ら [Benedens99] は3Dモデルの形状から求められた法線ベクタの集合を対象として、あるクラスの座標変換に対し頑強性を持つ透かしの埋め込み手法を述べた。これも一種の変換領域での埋め込みと言えるだろう。

位相の変更による埋め込み：3Dモデルのトポロジ、つまり頂点の結合関係を変更して埋め込む透かしもある。単純には、文字パターンの形にメッシュの大きさを変えたり、メッシュを切り絵のように文字などのパターンに切り抜くこともできる。また、3角ストリップを切り、その三角形の隣接関係にビット列を埋め込むこともできる [Ohbuchi98a, Ohbuchi98b]。

4.2.2 ポリゴンメッシュの形状属性への埋め込み

ポリゴンメッシュの形状に付随する属性、例えば頂点に付随する法線ベクトル、色、テクスチャ座標などや、面に付随する透明度、四面体に付随する温度や透明度などが埋め込みの対象となる。われわれはその一つのテクスチャ座標を変更してデータを埋め込む手法を提案した [Ohbuchi98b]。

4.2.3 ポリゴンのアニメーションデータへの埋め込み

MPEG-4 の facial animation parameters (顔アニメーションパラメタ群、FAPs) は顔全体で66自由度あり、たとえば上唇や目じりの位置などを小さな整数 (4-6bit) でパラメタ化して表現し、圧縮して転送することであらかじめ用意した顔の部品を動かすものである。Hartung ら [Hartung98] はこのFAPsのストリームの位相成分を偏重することで透かしを埋め込んだ。この手法で大変興味深いのは、透かしを埋め込んだ顔アニメーションをレンダリングして得られた2次元動画 (MPEG2圧縮) をFAPs抽出用の画像処理プログラムに入力しFAPsを抽出しすることで、レンダリング後の2次元動画画像から透かしの情報を取り出すことに成功した点である。

4.3 パラメトリック曲線・曲面への埋め込み

CADモデルの多くは、その形状の定義にポリゴンメッシュではなくパラメトリック曲線や曲面を用いる。また、殆どのCADモデルはその幾何形状やトポロジの変更を許さない。従って、ポリゴンメッシュを対象とし、形や位相を変える上述の電子透かし手法をそのままCADモデルに適用することは難しい。

以下ではCADモデルのパラメトリック曲面・曲線に対して透かしを埋め込む方針をいくつか述べる。以下、簡単のため主に曲線を対象に話しを進めるが、テンソル積曲面への拡張も容易に可能である。

内在する冗長性を使う：パラメトリック曲線では、厳密に同じ形状を複数の形状定義値で表現できる場合がある。この性質を使い、NURBS曲面・曲線を対象として有理線形関数による再パラメタ化を用いて情報を埋め込む手法が [Ohbuchi99] に述べられている。この手法の長所は厳密に幾何形状を保存し、さらにデータ量も保存することである。任意の再パラメタ化を加えると透かしが消え

るが、脆い透かしとして改ざんの検出等には使えるだろう。

冗長性を付加する： 冗長性を付加することで、データ量は増加するが形状は保存する埋め込みが実現できる。まず、複数の区間からなるパラメタ曲線、例えば非有理 B-Spline や NURBS 曲線に節点を挿入し、その節点の値などを使って情報を符号化し埋め込みむことが出来る。また、非有理な Bezier や B-spline 曲線を次数上げするとき、次数がどれだけ増えたか、どの区間に制御点を挿入したか、などに情報を埋め込むことができる。さらに、有理 Bezier や NURBS の次数上げを伴う再パラメタ化により情報を埋め込むことができる。これらの手法で導入された制御点や節点を、形状を厳密に保存しつつ除去することは困難のはずである。従って、ここで述べたアプローチは形状の保存を重んじる CAD データを対象に頑強な埋め込みの実現に向いていると予想される。

制約を緩める： 形状の厳密な保存は不要だがデータ量は保存したい場合の手法もいくつか考えられる。例えば曲面の制御ポリゴンをポリゴンメッシュとして扱い、ポリゴンメッシュの頂点座標値を変えて埋め込む手法をそのまま適用できる。また、曲線や曲面の制御点群には既に 1 次元あるいは 2 次元の順序がついているため、これら制御点座標値の列を単なる数値列として扱ってその数値を変更し、情報を埋め込むことが出来る。これらの手法はほとんど全てのパラメタ曲線に適用できる。また、節点ベクトルを単なる数値列として扱い、その値を変更して情報を埋め込むこともできる。制御点や節点ベクトルの数値を変更する場合、値をそのまま変更しても良いし、何らかの変換を施した領域でその係数を変更しても良い。変換領域で埋め込む手法は、例えば、増田が[増田 98]で提案した離散コサイン変換を用いた損失のある曲面形状圧縮手法と組み合わせられるだろう。

5. まとめ

本論分では、まず、3 次元モデルをメディアデータ型としてみたときに用意されるべき操作群について概観した。その後、インターネットを意識したときに特に重要な、圧縮と電子透かしの2つの操作についてその研究と実用化の現状を紹介した。圧縮はここ数年の集中的な研究でかなり実用化に近づいているが、電子透かしはまだまだに見える。だがこちらでも研究開発は急速に進むと思われる、目の離せない分野である。

参考文献

- [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>)
- [Benedens99] O. Benedens, Geometry-Based Watermarking of 3D Models, *IEEE CG&A*, pp. 46-55, January/February 1999.
- [Deering95] M. Deering, Geometric Compression, *Proceedings of the ACM SIGGRAPH '95*, pp. 13-20, 1995.
- [Fitzmaurice99] G. W. Fitzmaurice, R. Balakrishnan, G. Kurtenbach, Sampling, Synthesis, and Input Devices, *CACM*, Vol. 42, No. 8, 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.
- [Gumhold98] S. Gumhold, and W. Strasser, Real Time Compression of Triangle Mesh Connectivity, *Proc. ACM SIGGRAPH '98*, Orlando, Florida, USA, 1998.
- [Hoppe96] H. Hoppe, Progressive Meshes, *Proc. ACM SIGGRAPH '96*, pp. 99-108, 1996.
- [VRML] ISO/IEC 14772-1 Virtual Reality Model Language (VRML).

[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 Modeling: Fundamentals and Applications (GEO-6), pp. 296-307, Tokyo, Japan, December 1998.

[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

[増田99] 増田 宏, 大瀨 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7号, pp.1188-1195, 1999年3月号.

[松井98] 松井 甲子雄, 電子透かしの基礎, 森北出版, 東京都, 1998年8月.

[Ohbuchi97] R. Ohbuchi, H. Masuda, and M. Aono, Watermarking Three-Dimensional Polygonal Models, *Proceedings of the ACM Multimedia '97*, pp. 261-272, 1997.

[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, *Proc. Computer Graphics International '99*, pp. 180-187, 1999.

[Popovic97] J. Popovic and H. Hoppe, Progressive Simplicial Complexes, *Proceedings of the ACM SIGGRAPH '97*, pp. 217-224, 1997.

[Praun99] Emil Praun, Hugues Hoppe, Adam Finkelstein, Robust Mesh Watermarking, *Proc. ACM SIGGRAPH '99*, pp. 49-56, 1999.

[Rossignac99a] J. Rossignac, Edgebreaker: Connectivity compression for triangle meshes, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 5, No.1, January-March 1999.

[Rossignac99b] J. Rossignac and G. Taubin, organizers, 3D Geometric Compression, *ACM SIGGRAPH '99 Course Notes #22*, 1999.

[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.

[Virtue3D] Virtue 3D Inc., <http://www.virtue3d.com>

[Yeo99] B.-L. Yeo and M. M. Yeung, Watermarking 3D Objects for Verification, *IEEE CG&A*, pp. 36-45, January/February 1999.