

Skin-Off: 3次元ビデオ圧縮のためのテクスチャ指向型2次元平面展開法

Skin-Off: Texture-oriented 2-D Plane Mapping for 3-D Video Compression

学生会員 曾良洋介[†], 波部 斉^{††}, 松山隆司[†]

Yosuke Katsura[†], Hitoshi Habe^{††}, and Takashi Matsuyama[†]

Abstract 3-D video records dynamic 3-D visual events as is. Applications of 3-D video include wide varieties of human activities. For promoting these applications in everyday life, a standardized compression scheme for 3-D video is required. We developed a practical and effective scheme for representing and compressing 3-D video named "skin-off," in which both the geometric and visual information are efficiently represented by cutting a 3-D mesh and mapping it onto a 2-D array. Skin-off has much in common with geometry images proposed by Gu et al. However, while geometry images use only 3-D surface shape information to generate 2-D images, skin-off uses both 3-D shape and texture information to generate them. This enables us to achieve higher image quality with limited bandwidth. Experimental results demonstrate the effectiveness of skin-off.

キーワード: 3次元ビデオ, 圧縮, 符号化, 幾何情報, テクスチャ, Skin-off

1. ま え が き

計算機やGPU, ディスプレイなどの目覚ましい発展に伴い, いままでになかったタイプの映像技術が盛んに開発されている. そのひとつに「3次元ビデオ」¹⁾が挙げられる. 3次元ビデオは, 複数視点のカメラを用いることで撮影対象の形状, 色, 動きなど, 実世界をありのままに記録するメディアである. 今後, スポーツ選手の動作解析, 伝統舞踊のアーカイブや教育目的のコンテンツなどへの応用が期待される.

しかし, 3次元ビデオのように, 3次元世界を詳細に記述するためには膨大なデータが必要となる. そのため, 伝送の際には, 帯域の制限を受けてフレームレートの低下を引き起こし, 蓄積の際には有限な記憶領域を圧迫するなどの問題が生じる. これらの問題は, データ圧縮することで解決できるが, 3次元ビデオは比較的新しいメディアであるため, 従来の2次元動画圧縮法のように広く使われてい

る標準的な圧縮手法が存在しないのが現状である. そのため, 効率的な3次元ビデオ圧縮手法の確立が望まれており, 国際標準を決めるMPEGなどにおいても活発な議論が進められている²⁾³⁾.

本論文では, 3次元ビデオのための圧縮手法を提案する. ここで言う3次元ビデオとは, 時間経過とともに動的に変化する3次元メッシュと, その表面に対応付けられるテクスチャからなるものとする. この3次元ビデオに対し, 我々の提案する手法はまず, 3次元メッシュに適切な切れ目を入れて切り開き, 2次元平面に展開できるようにする. そして, 幾何情報とテクスチャ情報を2次元平面に展開し, 2次元画像へと変換し, 変換された2次元画像列を従来の動画圧縮手法を用いて圧縮する(図1参照). この手法は, 既に広く使われている2次元動画圧縮手法を利用でき, 既存技術との親和性が高いという利点がある.

3次元メッシュを2次元画像に変換して圧縮する手法は, GuらのGeometry Images⁴⁾でも提案されており, Briceñoらによって動画への適用も検討されている⁵⁾. しかし, これらの手法は幾何情報(3次元メッシュ)のみを効率的に表現・圧縮する手法である. これに対し, 前述した通り, 3次元ビデオとは3次元メッシュとテクスチャ情報によって構成される. 例え3次元メッシュが精度よく保存されていたとしても, そのメッシュに貼られるテクスチャの画質が低いものであれば, 結果として表示される映像は画質の低いものとなる. 逆に, 3次元メッシュに多少の歪みが生じ

2005年9月1日受付, 2005年11月28日再受付, 2005年12月28日採録

[†] 京都大学 大学院 情報学研究科

(〒606-8501 京都府京都市左京区吉田本町, TEL 075-753-4891)

^{††} 京都大学 大学院 工学研究科

(〒606-8501 京都府京都市左京区吉田本町, TEL 075-753-7465)

[†] Graduate School of Informatics, Kyoto University

(Yoshida-Hommachi, Sakyo-ku, Kyoto-shi, 606-8501, Japan)

^{††} Graduate School of Engineering, Kyoto University

(Yoshida-Hommachi, Sakyo-ku, Kyoto-shi, 606-8501, Japan)

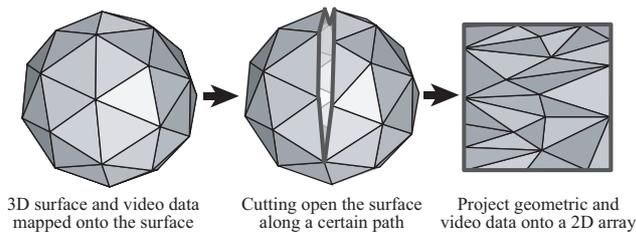


図 1 Skin-off アルゴリズムの概略
Overview of skin-off.

た場合でも、テクスチャが高画質であれば見た目にきれいな映像が得られると考えられる。

このように、3次元ビデオの圧縮を考える上では、最終的な映像品質に対して、幾何情報とテクスチャ情報がどのような影響を及ぼすかを考慮しなければならない。本論文の基本的アイデアは、映像品質に大きな影響を及ぼすテクスチャ情報を優先し、データ量に対する映像品質の効率を最大化する2次元平面展開を行うことにある。提案手法中で、3次元物体を2次元平面に切り開く処理がちょうど動物の皮を剥くようなイメージであるところから、本圧縮手法をSkin-offと呼ぶ。

本論文では、2節で本論文で扱う3次元ビデオを改めて定義して提案手法の基本的考え方を述べ、関連研究との相違を論じる。次いで、3節でSkin-offアルゴリズムについて詳述する。4節では実験を通してSkin-offの有効性を示し、最後の5節でまとめを行う。

2. 3次元ビデオとその圧縮

2.1 3次元ビデオ

本論文で対象とする3次元ビデオとは、時間とともに変化する、

- 3次元メッシュ

● 3次元メッシュの表面に対応付けられるテクスチャの二つの要素で構成されている。ここでの3次元メッシュは、3次元空間中の頂点集合とその接続関係を示すエッジ情報を指しており、テクスチャはRGBなどの色情報を指している。

このような3次元ビデオは、複数のカメラから得られた映像をもとに、視体積交差法をなど用いたりすることで獲得できる¹⁾。しかし、一般的には、このような方法で得られた3次元ビデオでは、各フレーム毎に形状が復元され、各時刻間で頂点の数や位置の対応がとれていないため、時間軸方向の相関がなく、圧縮効率が悪い。

これに対して、Skin-offでは、弾性メッシュモデル¹⁾を用いて、対象の形状と動きを同時に復元して得られた3次元メッシュを入力として用いる。この3次元メッシュでは、時刻間での頂点の対応がとられているため、頂点同士の接続関係は時間変化に対して不変となり、圧縮処理に対して非常に有利となる。

また、本論文で提案する手法は、一つの3次元物体に対

する圧縮手法であるが、複数の物体が存在する場合には個別に圧縮処理を行えば良い。

2.2 幾何情報とテクスチャ情報が映像品質に与える影響
前述したように、Skin-offアルゴリズムは、幾何情報とテクスチャ情報が映像品質に与える影響を考慮した点を特長としている。このアイデアを具体化するためには、幾何情報とテクスチャ情報を具体的にどのように定量化し、それぞれがどのように映像品質に影響を与えるかを考えなければならない。

幾何情報とテクスチャ情報の情報量の大小は、形状の複雑さと模様複雑さを、それぞれ評価尺度とできると考えられる。ここで、圧縮によってある領域の幾何情報、あるいはテクスチャ情報が失われたとき、他方の情報量の大小と映像品質との関係を考える。ここでは、議論を簡単にするために、それぞれの情報量が多い場合と少ない場合を考えることにする。

まず、幾何情報が失われた、すなわち形状に歪みが生じたとする。もし、その領域に充分細かなテクスチャが存在していたとすると、映像上でも形状歪みが目立ち、品質が劣化したと知覚される。逆に、有意なテクスチャが存在していなければ、若干の形状歪みは目立たず、品質の劣化はさほど感じられないと考えられる。次に、テクスチャ情報が失われ、ボケなどが生じた場合は、メッシュ形状にかかわらずそれが映像上に現れるため、品質の劣化につながる。

つまり、3次元ビデオの圧縮において、幾何情報およびテクスチャ情報と映像品質の関係は、

- 幾何形状に許される歪みは、その表面のテクスチャ情報の大小によって決まり、テクスチャ情報が多い程、幾何歪みが小さくなるようにしなくてはならない。

- テクスチャ情報は、対応する部分の幾何形状の複雑さとは関係なく、歪みが映像品質に影響する。そのためテクスチャ情報はできるだけ保持されなくてはならない。

となっていると言える。

2.3 関連研究

CGの分野を中心として、3次元ビデオの幾何情報の圧縮に関連する研究が行われている。Taubinらは、3次元幾何情報を構成する頂点座標と頂点接続情報の二つの情報をそれぞれ符号化する方法⁶⁾で圧縮を試みた。山崎らは、従来の2次元動画画像圧縮に用いられているブロックマッチングを3次元空間に拡張することで圧縮する方法⁸⁾を提案している。これらの手法は、幾何情報を圧縮する上では非常に有効な手法であるが、頂点座標と頂点接続情報を別個に扱わなくてはならず、それぞれに特別なコーデックが必要になる。

また1節で紹介したように、Guらは、3次元幾何情報を2次元矩形画像に変換することで、既存の圧縮手法を用いた圧縮を可能にしている⁴⁾。Briceñoらによって動画圧縮への応用も検討されている⁵⁾。これらの手法では、幾何的な歪みを評価するGeometric Stretchを最小化すること

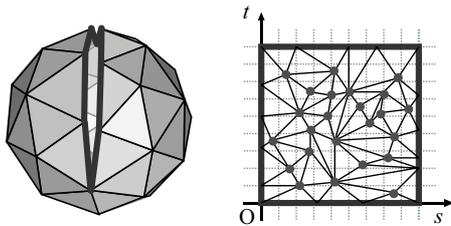


図 2 カットパスと平面展開
Cutpath and parameterization.

で、圧縮・復元後における幾何情報の損失を抑えており、テクスチャ情報は考慮されていない。

テクスチャの表現方法に関する研究も多く発表されている。Sander らは、法線情報、光学情報などのメッシュに付加される信号を復元することに特化した平面展開方法⁹⁾について考察している。Lévy らは、テクスチャをいくつかに切り分け、それらを一枚の画像に効率的に収める手法⁷⁾について述べている。

以上のように、CG 分野を中心に 3 次元メッシュやテクスチャを効率的に表現・圧縮する手法が数多く存在しているが、それらは幾何情報あるいはテクスチャ情報のみに着目しており、2.2 節で述べたとおり、3 次元ビデオの映像品質を考える上では不十分であるといえる。これに対し、本論文で提案する Skin-off は、2.2 節で論じた幾何情報とテクスチャ情報と映像品質との関係を考慮し、データ量に対する映像品質の効率を向上させるものである。

3. Skin-off アルゴリズム

前節までの議論を踏まえ、本節では我々の提案する 3 次元ビデオ圧縮手法 Skin-off の詳細なアルゴリズムについて述べる。

3.1 Skin-off アルゴリズムの概略

3 次元メッシュを 2 次元平面に展開する操作には、まず、3 次元メッシュを 2 次元平面に展開するために必要な「切れ目」を入れる必要がある。以降「切れ目」に相当するエッジの集合をカットパスと呼ぶ(図 2 の太線部分)。このカットパスを 2 次元平面上の輪廓に対応づけるに展開操作が行われるが、そこではさまざまな原因で歪みが生じ、映像品質の劣化につながる。それらを大別すると、

Type A カットパスの周辺が引き伸ばされることによる歪み、

Type B 任意形状のメッシュを 2 次元平面に投影することによる歪み、

に分けることができる。Type A は、カットパスが短いためにその周辺が極端に引き伸ばされるような場合や、カットパスが長いためにカットパスから離れた領域が押しつぶされるような場合を指している。Type B は、例えば、人間の腕のような尖った部分を無理に平面に投影するために生じる歪みである。カットパスの選択によって Type A の歪みを低減可能なのは明らかであり、Type B についても、

例えば、腕に沿ってカットパスを選べば歪みが低減できるように、カットパスの選択がこれらの歪みを大きく左右する。

したがって、本論文で提案する Skin-off では、カットパスの選択と更新に重点をおいて、最適な平面展開操作を求める。アルゴリズムの概要は以下ようになる。

Step1:初期カットパスの決定 与えられた 3 次元メッシュを 2 次元平面に展開するために必要なカットパスを決定する。ここでは、2 次元平面展開のために最低限必要なカットパスを求め、初期カットパスとする。

Step2:頂点配置の最適化 カットパスを用いて、頂点を 2 次元矩形平面へ展開し、矩形平面内でのメッシュ頂点(図 2 右で 示す点)の配置を最適化する。

Step3:カットパスの更新 より歪みが少ない展開を可能にするカットパスが存在しないか探索し、そのようなカットパスがあれば更新する。Step2 と Step3 の操作を繰り返し、歪みを示す評価値が収束すれば Step4 に移る。

Step4:矩形領域内格子点でのサンプリング 平面展開された矩形領域の適当な解像度の格子点で 3 次元ビデオデータをサンプリングする。幾何情報では 3 次元座標を、テクスチャ情報では、例えば、RGB 値を取得する。サンプリングされた値を 2 次元配列として保存する。図 2 右では、平面上に展開された 3 次元頂点座標データが、破線で示す格子点でサンプリングされ、2 次元配列に記録される様子を示している。

Step5:2 次元映像圧縮手法の適用 2 次元配列を画像とみなし既存の圧縮手法を適用する。

このアルゴリズムによって、3 次元メッシュの隣接する頂点は、2 次元矩形画像でも隣接するように配置されるので、頂点の隣接関係を陽に保持する必要がない。すなわち、Step4 までで得られる 2 次元配列で、3 次元ビデオのすべての情報が表現されている。これは 3 次元ビデオの圧縮を考える上で大きな利点となる。

上記のアルゴリズムの中の、3 次元メッシュに切り目を入れて平面に展開するというアイデアは、Gu らによる Geometry Images⁴⁾ と共通しているが、2.3 節で述べたように、Geometry Images では考慮されていなかった 3 次元ビデオのテクスチャ情報に着目し、限られたデータ量での映像品質を向上させている点が提案手法である Skin-off の大きな特長である。

なお Skin-off では、Step4 のサンプリングの影響で幾何情報とテクスチャ情報の整合性がとれなくなるのを防ぐために、両者は同じ頂点配置で表現されるものとする。

以降では、アルゴリズムの中心となる Step1 から Step3 の処理について述べる。

3.2 幾何情報・テクスチャ情報の定義

幾何情報・テクスチャ情報の情報量の大小を計る尺度を定義する。幾何情報に関しては、その形状の複雑さが情報量の大小を決定すると考えられる。つまり、メッシュ表面

の曲率の大小が幾何情報の情報量を表わすと考えられる。本論文では、Gu らが Geometry Images で用いた評価量 $C(f)$ を曲率を近似するものとして採用する。ここで f は 3 次元メッシュの面を指している。 $C(f)$ の詳細は付録 A 節で述べる。一方、テクスチャ情報の評価値としても様々なものが考えられるが、2.2 節で考察したように、テクスチャ情報とはその複雑さによって決まるものである。そこで本論文では、面 f のテクスチャ情報量 $D(f)$ を式 (1) のように定義した。

$$D(f) = \frac{1}{N_f} \sum_{p \in f} \sqrt{d_x(p)^2 + d_y(p)^2} \quad (1)$$

ここで、 p は面 f に含まれる画素、 N_f は面 f に含まれる画素数である。 $d_x(p)$ 、 $d_y(p)$ はそれぞれ p における x 、 y 方向の微分を表わす。つまり $D(f)$ は、面 f での空間微分値の平均強度である。

3.3 カットパスの決定

本節では、カットパスの決定について述べる。3.1 節で述べたアルゴリズムの中では Step1 と Step3 に相当する。3.1 で述べたように、2 次元平面展開によって生じる歪みは大きく 2 種類に分類できる。ここで、それらとテクスチャ情報・幾何情報との関係を考える。

TypeA の歪みに関しては、カットパス周辺のテクスチャ情報が小さければ、歪みの映像品質への影響を低減できると考えられる。TypeB の歪みに関しては、3 次元メッシュの曲率が大きければ、2 次元平面に投影したときの歪みが大きくなるが、もしそこに有意なテクスチャが存在しなければ、その歪みは映像品質の劣化につながらず、有意なテクスチャがあれば映像品質の劣化につながる。これを避けるためには、曲率が大きく、有意なテクスチャがある領域をカットパスで切り開けばよい。

以上をまとめると、カットパスに求められる条件は、

- 一般的に、カットパスはテクスチャ情報が少ない領域を通るほうがよい。

- 曲率が大きく、有意なテクスチャがある領域はカットパスで切り開いた方がよい。

となる。これらは矛盾しているようにみえるが、TypeB による歪みは、曲率が大きければ非常に大きくなるため、後者によって生じる TypeA の歪みは無視できると考えられる。この考え方に基づく Step1 と Step2 のアルゴリズムは以下ようになる。

(1) 初期カットパスの決定

Step1 では、任意形状の 3 次元メッシュを 2 次元に展開するために最低限必要なカットパスを探索するが、ここで選ばれたものが以降の処理の初期値となるので、最終結果にも大きな影響を及ぼす。よって、先に述べた条件を考慮しながら、2 次元平面展開に必要なカットパスを探索する。

ここで、カットパス周辺のテクスチャ情報量を定量化するために、各エッジ e に与えられるテクスチャ情報量 $D(e)$

を下式で定義する。

$$D(e) = D(f_1) + D(f_2) \quad (2)$$

ここで、 f_1 、 f_2 はエッジ e に接している二つの面である。具体的な処理は以下ようになる。

Step1.1 探索処理の初期値となる一つの面をランダムに選ぶ。

Step1.2 Geometry Images と同様の手法で平面展開に最低限必要なカットパスを探索する⁴⁾。具体的には、Step1.1 で選んだ面を除去したあとで、ただ一つの面にしか接しないエッジとそれが接する面を順に除去していき、最後に残ったものをカットパスとするものである。

Step1.3 得られたカットパスに含まれるエッジで $D(e)$ の和を計算する。

Step1.4 Step1.1 から Step1.3 までの処理を何回か行い、 $D(e)$ が最小となったもの、すなわち、テクスチャ情報が少ない領域を通るものを初期カットパスとする。

(2) カットパスの更新

カットパスの更新では、先に述べた二つの条件を満たすように、以下のアルゴリズムを用いる。

Step3.1 各面に対して、曲率 $C(f)$ とテクスチャ情報 $D(f)$ の積を計算し、それが最大となる面を求める。

Step3.2 Step3.1 で得られた面から現在のカットパスの間で、テクスチャ密度 $D(e)$ の和が最小となるような経路を探索し、新たにカットパスに付け加える。

このアルゴリズムによって、曲率とテクスチャ情報が極端に大きいところから、できる限りテクスチャ密度が低いエッジを通るカットパスが得られる。

3.4 矩形領域中での頂点配置の最適化

まず、カットパスに含まれる頂点を矩形の輪郭上に配置する。切り開きの際に、カットパスに含まれる頂点、エッジは複製される。Step4 のサンプリングをした後のデータで、3 次元形状を再構成したときに隙間が生じないように、カットパスに含まれる頂点は Step4 でサンプリングされる点に配置する。

次に矩形内部にある頂点の位置を最適化する。この頂点の配置によって幾何情報とテクスチャ情報のサンプリング間隔が決定される。2.2 節で述べたように、限られたデータ量の中で効率的に表現するためには、テクスチャ密度の高い部分で幾何歪みを少なくする必要がある。

(a) 最適化の条件

頂点配置の最適化では、以下の三つの条件を満たすように目的関数を設計する。

(1) テクスチャ情報 $D(f)$ の大きい部分は 2 次元平面上でも大きくマッピングされる。

(2) 隣接する面では平面上での面積が極端に異なることがないようにする。

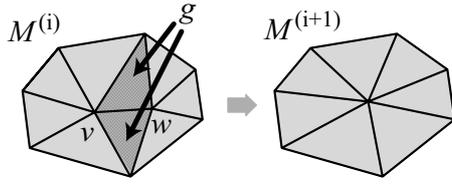


図 3 メッシュ簡略化のための頂点併合
Vertex contraction for mesh simplification.

(3) 各面の形状は3次元空間と2次元平面上でできる限り同じ形状とする。

(1) は面 f のサンプリング間隔を $D(f)$ によって変化させるための条件であり, (2) は隣接面でのサンプリング間隔が極端に異なることによって生じる歪みを防ぐための条件である。また, 3次元空間と2次元平面上で面の形状が変化した場合, 大きく引き伸ばされた方向のデータが無駄となるため, 条件 (3) を加えている。上記の条件を満たす目的関数 TG を具体的に下式で定義する。

$$TG = \sum_f ((\alpha \cdot TS(f) + 1) \cdot GS(f)) \quad (3)$$

上式中の $TS(f)$ は上記の条件 (1)(2) を満たすための項であり,

$$TS(f) = \frac{A_{3D}(f)}{A_{2D}(f)} \sum_{f_n \in F} w(f, f_n) D(f_n) \quad (4)$$

で定義する。 $A_{3D}(f)$, $A_{2D}(f)$ はそれぞれ3次元中, 2次元中での面 f の面積である。 F は面 f 自身とその近傍の面を含む集合, $w(f, f_n)$ は面 f と面 f_n の距離が近い程大きな値をとる関数である。テクスチャ密度 $D(f)$ が大きい面を平面上で大きく配置すると $TS(f)$ は小さな値をとる。 $GS(f)$ は条件 (3) を満たすための項である。 $GS(f)$ は Geometry Images でも用いられていた Geometric Stretch⁴⁾ であり,

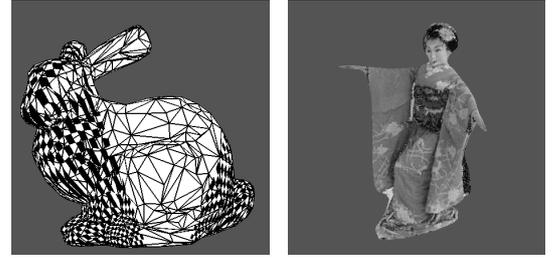
$$GS(f) = \sqrt{((\partial S / \partial s)^4 + (\partial S / \partial t)^4) / 2} \quad (5)$$

で定義される。ここで S は3次元空間への頂点の対応付けを表わす関数であり, s, t は2次元平面での座標を表わす。 $GS(f)$ は2次元平面への展開でのメッシュ形状の変形の度を表わしている。

TG は基本的に $TS(f)$ と $GS(f)$ の積で定義されるが, 式 (3) 中で $TS(f)$ の項に定数 1 を加えているのは $D(f)$ が 0 となる面でも $GS(f)$ を考慮できるようにするためである。また, 係数 α によってテクスチャ情報 $TS(f)$ を重要視する割合を制御でき, $\alpha = 0$ のときは Geometry Images と同じ条件となる。

(b) 最適解探索のためのメッシュ簡略化

矩形領域内での頂点配置の自由度は非常に高く, 局所解に陥ってしまう可能性が高い。これを防ぐために, メッシュを簡略化し, まず始めに簡略化されたメッシュで大まかな最適化を行い, 順次メッシュを細かくしながら, 最適な頂



(a) Stanford Bunny (b) 舞妓

図 4 実験に用いたデータ
Experimental data.

点配置を得る方法を採用する。同種の手法は, Geometry Images でも用いられているが, テクスチャ情報を考慮していないため拡張が必要になる。

(c) アルゴリズム

以下にメッシュ簡略化を含む Step2 の処理を述べる。

Step2.1 3次元メッシュの $M^{(i)}$ の頂点 v とそれに隣接する頂点 w のすべての組合せに対し, v と w のいずれかに接する面 f のテクスチャ密度の最大値と最小値の差 $D_{max} - D_{min}$ を求める。この差が, 注目する頂点付近のテクスチャ密度の一様性を表わしている。

Step2.2 $D_{max} - D_{min}$ が最小となった頂点の組 v, w を図 3 のように一つの頂点に併合し, 新たなメッシュ $M^{(i+1)}$ を得る。なお, カットパスに含まれるエッジは併合によって消去されないようにする。また, 頂点 v, w に隣接し, 併合によって消去されずに変形した面 f のテクスチャ密度を式 (6) によって定義する。

$$D(f') = \frac{A(f) \cdot D(f) + \sum A(g) \cdot D(g)}{A(f) + \sum A(g)} \quad (6)$$

ここで, g は消去される面, A は面積を表わす。

Step2.3 併合可能な面がすべて併合され, カットパスに含まれない点が一つになるまで Step2.1, Step2.2 を繰り返す。最終的に得られたメッシュを $M^{(N)}$ とする。

Step2.4 最も簡略化されたメッシュ $M^{(N)}$ に対して, 目的関数 TG を最小化するように頂点配置を最適化する。

Step2.5 $M^{(N-1)}$ については, $M^{(N)}$ を最適化した頂点配置を初期値として, 同様の最適化を行う。この処理を $M^{(0)}$ まで繰り返す。

Step2.6 $M^{(0)}$ で得られた最適化における TG の値があらかじめ定めた閾値を下回った場合, Step1 から Step3 を終了し, Step4 に進む。下回らなかった場合, Step3 に進み, カットパスの更新を行う。

4. 評価実験

本論文で提案した3次元ビデオ圧縮アルゴリズムの有効性を検証するための実験を行った。実験では, 図 4 に示すテクスチャを張った Stanford Bunny と, 実写から得られた舞妓のデータを用いた。

実験では, テクスチャ情報を考慮した平面展開の有効性

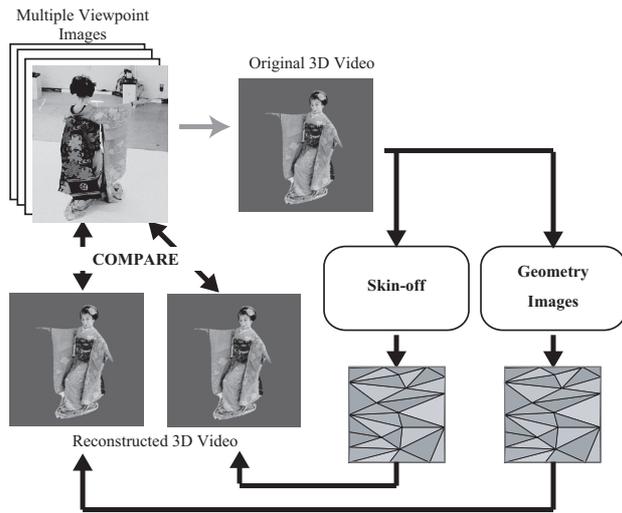


図 5 実験手法
Experimental method.

表 1 実験に用いたデータのデータ量
The amount of data for experiments.

| | Stanford Bunny | 舞妓 |
|--------------|----------------|-----------|
| 入力データ | | |
| 頂点数 | 772 | 998 |
| 面数 | 1502 | 2000 |
| 2次元平面展開後のデータ | | |
| 頂点数 | 65 × 65 | 65 × 65 |
| テクスチャ画像の大きさ | 512 × 512 | 512 × 512 |

を確認するために、3で挙げた Step1 から Step4 までの処理を行った。そして、平面展開された幾何とテクスチャを用いて3次元物体を再構成し、ある視点から見た画像を作成した。

比較対象としたのは、(1) Step1 から Step4 までの処理を行ったもの（提案手法：以下では Skin-off と呼ぶ）、(2) Skin-off の処理からテクスチャを考慮した部分を省略したもの（Geometry Images に相当する）の二つとした（図5）。これによって、提案手法の主たるアイデアである、2次元平面展開時のテクスチャ情報の利用の有効性が明らかになると考えられる。実験で用いたデータのデータ量を表1に示す。

まず、図6に Stanford Bunny に対して、最終的に得られたカットパスの様子を示す。図中の灰色の太線がカットパスに相当する。この図を見ると、幾何情報の評価のみでは切り開かれることのなかった尻尾の部分まで切り開かれているのがわかる。また、よりテクスチャの少ない部分を通るように切り開かれている様子もわかる。

次に図7に Stanford Bunny を平面展開した様子を示す。図中の (a),(b) が展開されたメッシュを示し、(c),(d) が展開されたテクスチャを示している。(d) と (c) を比べると、Skin-off アルゴリズムによって、テクスチャ密度の高い部分がより大きくマッピングされていることがわかる。

最後に、特定の視点から撮影対象を見たの時の画像を示した。図8は Stanford Bunny の尻尾の部分である。Ge-

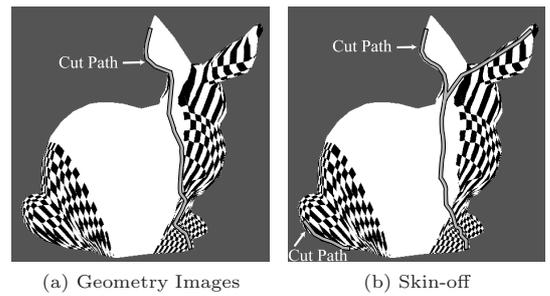


図 6 Stanford Bunny : カットパス
Stanford Bunny: Selected cutpath.

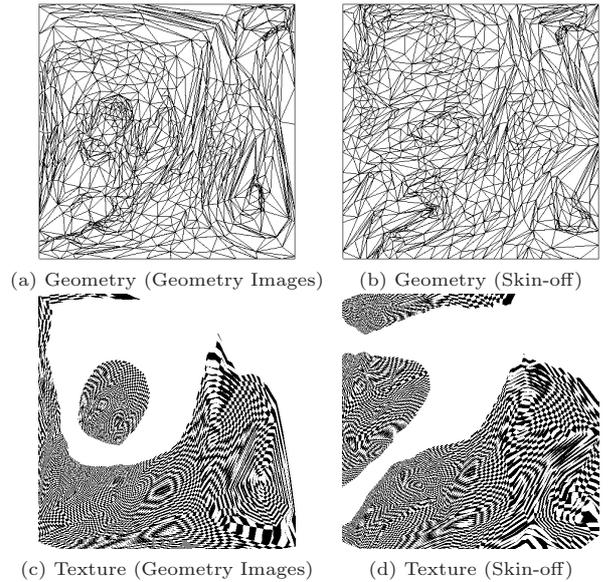


図 7 Stanford Bunny : 展開図
Stanford Bunny: Unfolded 2D array.

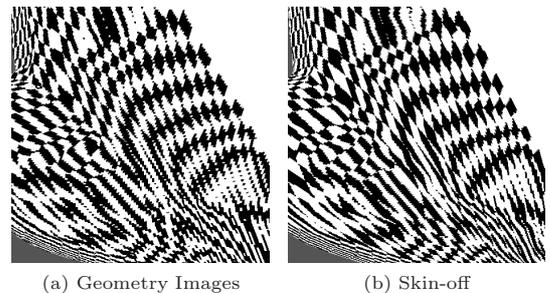


図 8 Stanford Bunny : 結果画像
Stanford Bunny: Result images.

ometry Images の場合だと、本来直線であるべきところに段差が見られていたが、Skin-off アルゴリズムを用いることによって、画質が改善されているのがわかる。これは、前述した尻尾部分を通るカットパスが、画質の向上に対して有効に作用したことを示している。

次に、実画像から生成された舞妓の3次元ビデオに対する実験結果を示す。図9に最終的に得られたカットパス、図10に平面展開されたメッシュと、テクスチャ情報を示す。また、特定の視点から得られた画像を図11、12に示す。

図9を見ると、Geometry Images では舞妓の頭部が切り開く必要がないと判断されているが、Skin-off アルゴリズムでは、曲率が大きくかつ有意なテクスチャがある頭部

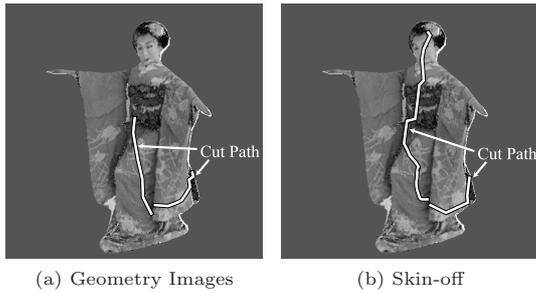


図 9 舞妓：カットパス
Maiko: Selected Cutpath.

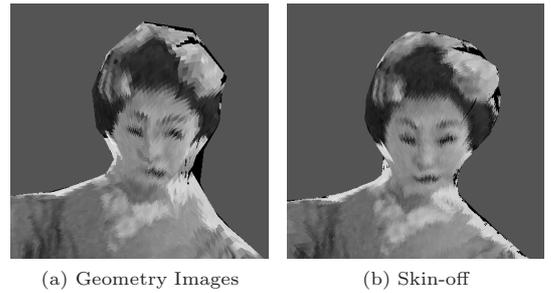


図 11 舞妓：結果画像 (顔)
Maiko: Result images(face).

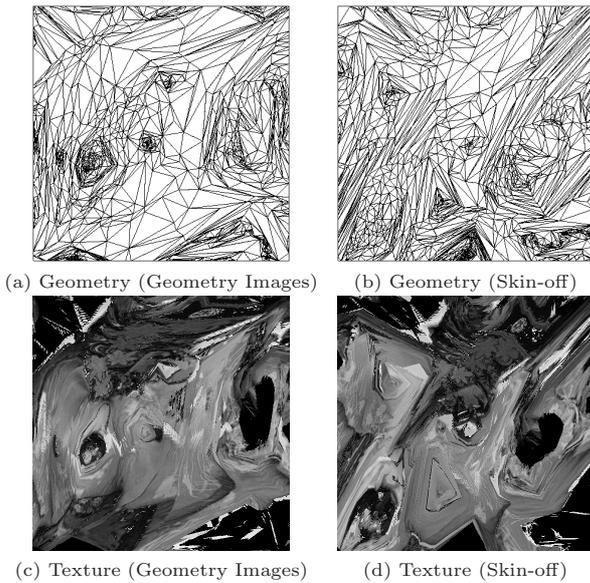


図 10 舞妓：展開図
Maiko: Unfolded 2D array.

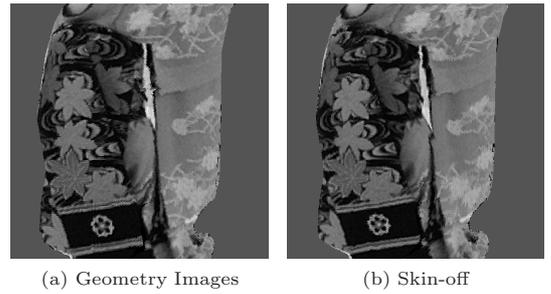
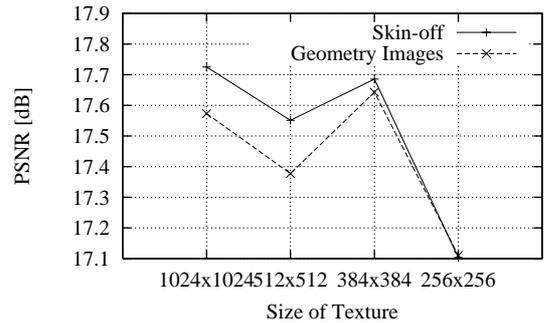


図 12 舞妓：結果画像 (帯)
Maiko: Result images(Obi).

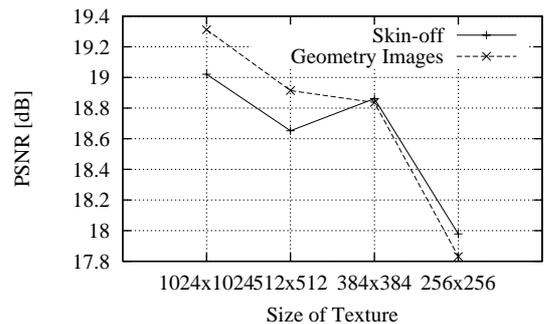
にカットパスが引かれている．そのため図 11 に示した舞妓の顔の画像を比べると，Geometry Images では首の左側に大きな歪みが生じ，顔全体にも縦方向に引き伸ばされているのに対し，Skin-off アルゴリズムを用いた場合にはそのような歪みが見られない．

また，図 12 に示した帯の部分の画像は，テクスチャ情報が大きく，曲率が低い部分である．このような部分では，カットパス選択による影響が少ない．Skin-off を用いたときの顔のように，カットパスを入れて画質が改善されている部分があると，限られたデータ量の中ではそのしわ寄せを受けて情報が失われる可能性がある．しかし，図 12 からわかる通り，最適化の際にテクスチャ情報を用いることでその情報を失わずに，幾何情報のみを用いた場合と同等の画質となることが見てとれる．

最後に，PSNR の算出によって定量的な評価を試みる．Skin-off や Geometry Images は，非可逆な平面展開を行うため，その結果得られたデータは，幾何情報とテクスチャ双方の誤差を含んでいる．このうち注意が必要なのは幾何情報であり，例えば，画像上を平行移動した場合は，人間の目にはその差が目立たないが，画素単位の差分を求める PSNR では差が非常に大きくなることが考えられる．すな



(a) 顔 (図 11)



(b) 帯 (図 12)

図 13 PSNR 計算結果
PSNR values.

わち，本手法の評価の際には，PSNR の評価結果と主観評価の結果が大きく異なることが考えられる．

図 11, 12 を求めたデータについて PSNR を求めた結果を図 13 に示す．平面展開後のテクスチャ画像の大きさを 4 通りに変化させ，それぞれについて PSNR を求めた．PSNR 算出時には，平面展開後再構成した 3 次元ビデオから実際のカメラ位置から見た画像を生成し，カメラから得られた実画像と比較した．図 13(a) をみると，概ね Skin-

off の画質が良くなっているが、テクスチャ情報が極端に少なくなるとその差が小さくなっている。(b) をみると逆に Geometry Images の方が良い場合がある。この原因としては、Geometry Images ではテクスチャがぼけたようになっているため、帯の部分に生じた幾何歪みによる PSNR の低下が少なくなっていることが考えられる。先述のとおり、PSNR は評価尺度として充分ではなく、新たな定量評価手法の確立や、主観評価と組合せた評価が必要である。

5. む す び

本論文では、3次元ビデオ圧縮のための平面展開手法 Skin-off を提案した。Skin-off では、3次元メッシュとその表面に対応づけられたテクスチャを2次元平面に展開する。これによって、通常の2次元平面映像用のコーデックを用いて、3次元ビデオの情報を圧縮することができ、実用面での利点も大きいといえる。

3次元メッシュやテクスチャを圧縮する手法は数多く提案されているが、その多くは幾何情報とテクスチャ情報のいずれかのみを扱っている。これに対し、Skin-off では、幾何情報とテクスチャ情報が映像品質に及ぼす影響を考慮し、最適な平面展開を求める。最終的に映像を鑑賞するときの品質には、テクスチャ情報が大きく影響すると考えられるため、Skin-off では、テクスチャの情報量が大きいところは、その劣化を抑えるように平面展開操作を決定している。高精細な3次元ビデオの圧縮を考える際には、このような考え方が不可欠であり、4節での実験結果がそのアイデアの妥当性を示していると言える。

論文中で述べたとおり、平面展開を行う際には、カットパスの選択が非常に重要になる。本論文で述べたアルゴリズムは、曲率やテクスチャ密度などの局所的な特徴量を用いて、最適なカットパスを探索している。実験結果では、それでも十分な効果があることが示されているが、今後、物体の全体的な形状などの大局的な情報を用いることで、効率がより向上することが期待される。

はじめに述べたように、3次元ビデオは次世代映像メディアの中核となる可能性を秘めている。3次元ビデオの実用化を推し進めるためには、効率的かつ実用的な圧縮手法の確立は不可欠であり、そのために今後も検討を続けていく。

本研究は、文部科学省「知的財産の電子的な保存・活用を支援するソフトウェア技術基盤の構築」事業の「大型有形・無形文化財の高精度デジタル化ソフトウェアの開発」プロジェクトの補助を受けた。実験で使用した Stanford Bunny は The Stanford 3D Scanning Repository のものである。

〔 文 献 〕

- 1) T.Matsuyama, X.Wu, T.Takai, S.Nobuhara: Real-Time 3D Shape Reconstruction, Dynamic 3D Mesh Deformation, and High Fidelity Visualization for 3D Video. CVIU, **96**, 3, pp.393-434 (2004)
- 2) ISO IEC JTC1/SC29/WG11(MPEG): "Report on 3DAV Exploration". WG11 document N5878, Trondheim, Norway.

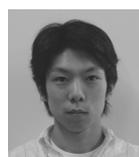
- 3) ISO IEC JTC1/SC29/WG11(MPEG): "Applications and Requirements for 3DAV". WG11 document N5326, Awaji, Japan.
- 4) X.Gu, S.J.Gortler, H.Hoppe: Geometry Images. ACM SIGGRAPH 2002, pp.355-361 (2002)
- 5) H.M.Briceño, P.V.Sander, L.McMlimman, H.Hoppe: Geometry Videos: A New Representation for 3D Animations. ACM SIGGRAPH 2003, pp.136-146 (2003)
- 6) G.Taubin, J.Rossignac: Geometric Compression Through Topological Surgery. ACM Transaction on Graphics, **17**, 2, pp.84-115 (1998)
- 7) B.Lévy, S.Petitjean, N.Ray, J.Maillot: Least Squares Conformal Maps for Automatic Texture Atlas Generation. Proceedings of ACM SIGGRAPH 2002, pp.362-371 (2002)
- 8) 山崎俊彦, 韓昇龍, 早瀬和也, 相澤清晴: 3D ビデオの圧縮. 第1回デジタルコンテンツシンポジウム (2005)
- 9) P.Sander, S.Gortler, J.Snyder, H.Hoppe: Signal Specialized Parametrization. Proc. 13th Eurographics Workshop on Rendering 2002, pp.87-100 (2003)
- 10) M.S.Floater: Parametrization and smooth approximation of surface triangulations. CAGD 14, pp.231-250 (1997)

付 録

A. 幾何情報の情報量 $C(f)^{4)}$

ある3次元メッシュとそれに対するカットパスが得られているものとする。そのメッシュをまず、Floater の提案した頂点配置最適化方法¹⁰⁾を用いて、各頂点を円内に配置する。矩形ではなく円に頂点を配置するのは、輪郭部における歪みの影響を少なくするためである。

このFloater の最適化では、エッジに対して重み付けした張力を与えることで、頂点配置を決定する。その後、それぞれの三角形パッチに対する歪みを Geometric Stretch で評価した値 $GS(f)$ を幾何情報の情報量 $C(f)$ とする。Floater のアルゴリズムによる頂点配置では、曲率の高い部分に歪みが大きくでるため曲率の評価値として用いることができると言える。



かつら ようすけ
曾良 洋介 2003年、京都大学工学部電気電子工学科卒業。2005年、京都大学大学院情報学研究所修士課程了。現在、同博士後期課程在学中。コンピュータビジョン、3次元ビデオの研究に従事。学生会員。



はべ ひとし
波部 斉 1997年、京都大学工学部電気工学第二学科卒業。1999年、京都大学大学院工学研究科電子通信工学専攻修士課程了。同年、三菱電機(株)入社。2002年、京都大学学術情報メディアセンター助手。現在、京都大学大学院工学研究科電子工学専攻助手。コンピュータビジョン、3次元映像メディアの撮影・伝送・表示に関する研究に従事。



まつやま たかし
松山 隆司 1976年、京都大学大学院修士課程了。京都大学助手、東北大学助教授、岡山大学教授を経て1995年より、京都大学大学院工学研究科電子通信工学専攻教授。現在、同大学院情報学研究所知能情報学専攻教授。2002年、学術情報メディアセンター長、京都大学評議員。2005年、情報環境機構長。工博。画像理解、人工知能、分散協調視覚、3次元ビデオの研究に従事。最近は「人間と共生する情報システム」の実現に興味を持っている。