

3次元Huモーメント不変量を用いた時系列ボリュームデータの圧縮

服部 雄一[†] 中澤 篤志^{†,††} 町田 貴史^{†,††} 竹村 治雄^{†,††}

[†] 大阪大学大学院 情報科学研究科

^{††} 大阪大学 サイバーメディアセンター

あらまし 時系列3次元データは、応用範囲が広く今後の発展が期待される研究分野である。この時系列3次元データはデータサイズが非常に大きく、これらを伝送および保存する際は圧縮処理が必要不可欠となる。本稿では、データの時間的冗長性を削減することを目的として、3次元に拡張したHuモーメント不変量を用いた時系列ブロックマッチングによる圧縮法を提案する。この手法では、従来手法では考慮されていなかった物体の回転運動を検出することで、圧縮率の向上を図っている。

キーワード ボリュームデータ、圧縮、ブロックマッチング、動き検出、不変量

Compression of Time Sequence Volume Data using 3D Hu Invariant Moments

Yuichi HATTORI[†], Atsushi NAKAZAWA^{†,††}, Takashi MACHIDA^{†,††}, and Haruo TAKEMURA^{†,††}

[†] Graduate School of Information Science and Technology, Osaka University

^{††} Cybermedia Center, Osaka University

Abstract Time sequence 3D data has a broad range of application and is a research area expected for the future development. The time sequence 3D data has such a huge data size that data compression is essential when it is transmitted and saved. This paper proposes a compression method by using the time sequence block-matching with extended 3D Hu invariant moment for the purpose of reduction of the data time redundancy. In this method, the data compression ratio is improved by detecting the rotational motion of an object that is not considered in the existing method.

Key words Volume data, Compression, Block matching, Motion detection, Invariant value

1. はじめに

時系列3次元データは、動的3次元オブジェクトのリアルな再現を行うことができるメディアとして近年、注目を集めている。特に、人間の身体動作の3次元データは、コンピュータ・アニメーション作成や運動動作解析、人間工学、生体力学、また教育学などの幅広い分野での応用が考えられる [1][2][3]。これは、実際の人間の動きをキャプチャし、時系列3次元データとして復元するものであり、現実の人間の身体動作をコンピュータによってリアルに再現することができる。

このような時系列3次元データの問題は、データサイズが大きくなることである。時系列3次元データは、フレームレートが大きくなるほどデータ量も大きくなり、数秒の身体動作データがギガバイト単位になる場合もある。このままでは伝送やデータの保存にも不都合である。これに対し、これまでにいくつかのデータ圧縮の手法が提案されてきた。これらは大きく「フレーム内圧縮」と「フレーム間圧縮」の2つに分けられる。フレ

ーム内圧縮は単一フレーム内での類似性を利用した圧縮であり、フレーム間圧縮は時間的に連続した（もしくは近傍の）フレーム間データの類似性を利用した圧縮である。2次元画像と同様、フレーム間圧縮により大幅なデータ量の削減が期待できる。

フレーム間圧縮では、データ内のオブジェクトの動きを追い、その動き情報と直前フレームとの差分情報を抽出することでデータ量の削減を行う。すなわち、オブジェクトの動きをどのようにして追うかが重要になる。従来の3次元データ圧縮手法においても動き検出は工夫されていたが、いずれも対象物体が3次元空間内を平行移動することを前提としている。しかし実際には、物体が3次元空間内で全く向きを変えずに移動を行う可能性は低く、いくらかの回転移動を含むものと考えられる。

そこで本研究では、画像内のオブジェクトの平行移動や回転移動の影響を受けない不変量の一つであるHuモーメント不変量を3次元に拡張して用いる手法を提案する。この不変量を用いてデータ内の類似部分を探索し、動き検出を行う。

2. 関連研究

本章では、時系列 3 次元データの圧縮に関連する研究を紹介する。

a) Geometry Image

Geometry Image は、Gu らが提案した 3 次元のメッシュデータ（静止画）を圧縮する手法である [4]。メッシュデータを 2 次元正方形テクスチャデータに変換することで圧縮を行う。まず 3 次元データをメッシュの特定の辺に沿って切断し、展開する。このときの切開パスは、展開図にした際に最も歪曲が小さくなるような経路を選択する。切開パスに沿って切り開いた展開図形の辺が、そのまま正方形テクスチャの 4 辺になる。次に、テクスチャ内のどの座標にどのメッシュ頂点を置くかを定める。境界部においては、テクスチャの境界に沿って切開パスのルートを割り当てていき、内部においては、図形的な「張度」が最小になるように配置を行う。張度とは図形の各部分における伸張率であり、全体の張度が大きいほど元の図形からの歪みが大きい。続いて、テクスチャの各点の元図形における空間座標を RGB 値に対応させ、色データを付ける。最後に、色の付いた平面画像をウェーブレット変換で圧縮する。

b) MPEG アルゴリズムを用いた圧縮

森下らの提案する手法は、MPEG アルゴリズムの 3 次元化を行うことにより時系列ボリュウムデータの圧縮を行っている [5]。フレーム内の 3 次元離散コサイン変換し量子化を行う。続いてつの I ピクチャを 2 次元動画とみなして 2 次元 MPEG を適用し、3 次元 I ピクチャとする。次にフレーム内レートコントロールを行う。レートコントロールとは、各フレームの情報量に合わせて、割り当てるビット量を調整することである。この手法では、あるフレームを符号化した際のマクロブロック毎の量子化スケールコードをグラフ化し、その中間値を保存して、次に同一のピクチャタイプの時に、スケールコードの初期値としてその値を用いる。時系列ボリュウムデータには似た画像が多いことを前提とした手法である。この手法では、各種フレーム内圧縮は行っているが、フレーム間圧縮は行っていないため、時間的冗長性の排除はできていないと考えられる。

c) 拡張ブロックマッチングによる 3 次元ビデオの圧縮

韓らの提案する手法は、森下らの手法と同じく MPEG アルゴリズムをベースにしているが、こちらはフレーム間圧縮も行っている [6]。圧縮対象となる 3 次元ビデオは 1 フレームずつ独立にモデルが生成されるため、隣接フレーム間の頂点数や接続関係が一定ではない。そこで、この手法では 2 次元動画圧縮で標準的に用いられるブロックマッチングを 3 次元空間に拡張して行っている。隣接するフレーム内で最も類似度の高いブロックを探索し、動きベクトルを抽出する。ここで、類似度の高いブロックとは、各メッシュ面の法線ベクトルの類似度の高さから判定する。そして、ブロック毎の予測誤差を量子化・符号化することにより効率的な圧縮を行う。

d) Skin-off アルゴリズム

波部らの提案する手法では、3 次元データのフレーム内圧縮を行う [8]。これは、Geometry Image に類似したアルゴリズム

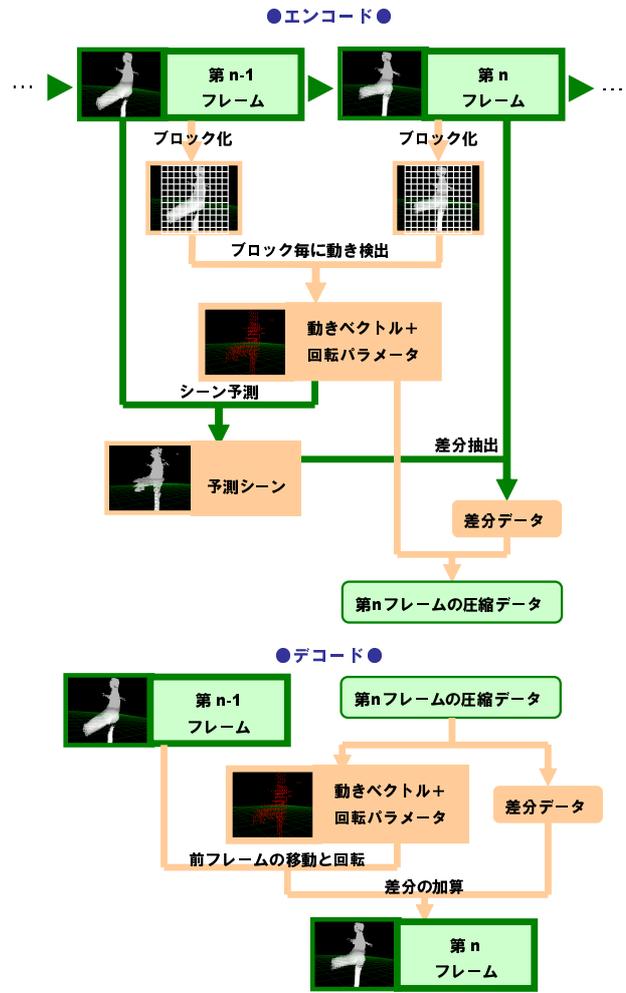


図 1 手法全体の流れ

である。Skin-off 法を用いて、与えられた 3 次元メッシュデータに対してある切れ目を入れて平面上へ展開し、3 次元メッシュの幾何情報を平面上で表現している。この手法は、3 次元の幾何情報に加え、メッシュ表面のテクスチャ情報も 2 次元平面上に展開する。物体の幾何形状は動的に変化するのに対し、その模様は変化しないことが多いため、テクスチャ情報は時間的冗長性が高い。すなわち、テクスチャ情報の圧縮は、3 次元ビデオの圧縮において重要な意義がある。

このように、3 次元データ圧縮には多くの先行研究があり一定の成果を上げているが、多くはメッシュデータを対象とした圧縮であり、内部密度情報を持つボリュウムデータを対象としたものは少ない。森下らの手法はボリュウムデータを対象としているが、時間的冗長性を排除できていないため、本稿ではボリュウムデータを対象としたフレーム間圧縮を提案する。

3. 提案手法

3.1 アルゴリズムの概要

まず本手法の全体の流れを示す。

エンコードはフレーム単位で行う。動画全般において、場面転換などが無いと仮定すれば、急激な画像の変化はほとんど考えられないため、連続するフレームの間は画像が類似する

と仮定できる．ここで、直前のフレームを使用して、現在のフレームの予測画像を生成することを考える．

被写体が完全に静止している場合、フレーム間の予測を行うことで差分は0になる．被写体がある程度の動きを伴う場合、あるフレームの被写体の含まれる部分を一部切り出し、それを動いた量だけずらして、それを次のフレームの画像として予測することができる．これを動き補償フレーム間予測という[9]．これは、被写体の動きを剛体の平行移動でモデル化した予測で、このモデルが適合するシーンに対しては圧縮効率が向上する．

図1に、第 n フレームのシーンを圧縮する場合の処理を示す．まず、第 n フレームのデータを $L \times L \times L(\text{voxel})$ サイズのブロックに分割する．そして、ブロック毎に動き検出を行う．これは、ブロック内部に含まれるボクセルのパターンに最も近い部分を前フレームから探し出し、そのブロックがどの位置から動いてきたかを推定する処理である．前フレーム内での探索範囲は「動き補償空間」と呼び、被写体が連続フレーム間で動く範囲を考慮して決定する．また、最も近い部分の探索は3次元に拡張したHuモーメント不変量[10]を比較して行う．この動き検出により、各ブロックの動き情報(動きベクトルと回転パラメータ)が得られる．

続いて、第 $n-1$ フレームのシーンと動き情報から、第 n フレームの予測シーンを作成する．第 $n-1$ フレームのシーンの最も類似した箇所を移動・回転させ、それらを合成して第 n フレームのシーンに近いものを作る．

そして、得られた予測シーンと第 n フレームのシーンの各ボクセル値の差分を抽出する．差分データは「予測誤差」ともいって、予測シーンの精度が高ければ高いほど差分値の合計は小さくなるため、データの圧縮効率が上がる．図1の通り、この「差分データ」と先ほどの動き情報(「動きベクトル+回転パラメータ」)をまとめて、第 n フレームのシーンの圧縮データとして出力する．

続いてデコードについて説明する．これは最初のフレームから順に行う．図1では第 n フレームのシーンを復元する場合を考える．まず、圧縮データから動きベクトル・回転パラメータを取り出し、それらの情報によって第 $n-1$ フレームのシーンの各部分を移動・回転させ、第 n フレームの各ブロックに格納する．続いて、圧縮データから差分データを取り出し、先ほど求めた各ブロックのボクセル値に差分値を加える．これで、第 n フレームのシーンを復元することができる．

3.2 Huモーメント不変量を用いた動き検出

エンコードで用いる動き検出アルゴリズムについて述べる．

3.2.1 時系列ブロックマッチング

ブロックマッチングとは、ブロック単位でパターンの照合を行うアルゴリズムである．一定のサイズのブロックをテンプレートとして、似たブロックがないか適当な探索範囲の中を探し、最も似ている部分を返す．

本手法では、各ブロックに対して以下の処理を行う．第 n フレームのシーン内の $L \times L \times L(\text{voxel})$ サイズのブロック(中心座標 (x, y, z))と、第 $n-1$ フレームのシーンの (x, y, z) を中心とした $3L \times 3L \times 3L$ サイズの動き補償範囲内の全ての

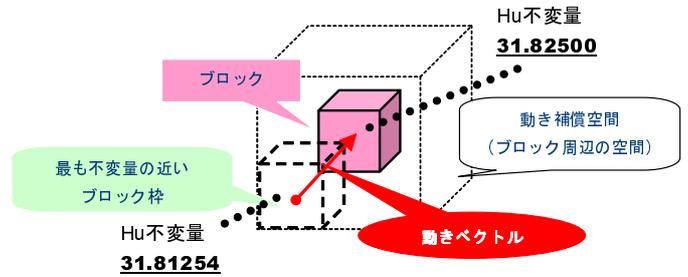


図2 Huモーメント不変量を用いた動きベクトル検出

$L \times L \times L$ サイズの部分とでブロックマッチングを行い、最も類似している位置 $(x_{sim}, y_{sim}, z_{sim})$ (ブロック枠の中心座標)を見つける．そして、 $(x - x_{sim}, y - y_{sim}, z - z_{sim})$ が、求める動きベクトルである．

本手法では、類似位置の探索に3次元Huモーメント不変量を用いる．図2にその概要を示す．各ブロックにおいて式(10)で表される3次元Huモーメント不変量 φ を求め、前フレームの動き補償範囲の中で同サイズの全てのブロックの内接球内の不変量値と比較し、最も近いブロックからのベクトルを動きベクトルとして検出する．

$$(v_x, v_y, v_z) = \arg \min_{(i,j,k)} |\varphi_n(x, y, z) - \varphi_{n-1}(x+i, y+j, z+k)| \quad (1)$$

3.2.2 Huモーメント不変量

Huモーメント不変量は、1つの画像に対して1つ存在する値で、画像内のオブジェクトの平行移動や回転移動、およびスケール変化の影響を受けない量である．画像内の図形が1つだけで、かつ、同じ形状であれば、それが画像のどの位置にあり、どの向きを向いていて、また、どのような大きさで表示されていようと、Huモーメント不変量は変わらない．また、2つの画像の被写体が合同もしくは相似に近い状態で写っていれば、不変量は近い値になる．

$f(x, y)$ を座標 (x, y) における画素値とする．画素値を元に、まずは画像モーメント m_{pq} を求める．

$$m_{pq} = \sum_{x,y} x^p y^q f(x, y) \quad (2)$$

画像モーメントは、画像の原点 $(0, 0)$ を中心とした画素の分散値である．大きな値を持つ画素値が原点から離れた座標に散らばるほど、画像モーメントは大きな値となる．ここで、添字の p, q は、それぞれX軸方向、Y軸方向に対する重みを表しており、 p の値が大きくなればX軸方向への分散に対する重みが、 q の値が大きくなればY軸方向への分散に対する重みが増した値となる．

続いて、重心 (\bar{x}, \bar{y}) を以下の式によって求める．

$$(\bar{x}, \bar{y}) = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right) \quad (3)$$

\bar{x} はX軸方向の重心、 \bar{y} はY軸方向の重心である．

この重心座標をもとにして、重心モーメント(重心を考慮し

た画像モーメント) μ を以下の式で求める .

$$\mu_{pq} = \sum_{x,y} (x - \bar{x})^p (y - \bar{y})^q f(x, y) \quad (4)$$

これは、原点からの距離ではなく、重心からの距離を考慮して求めるモーメントである . すなわち、この値は重心を考慮した画素の分散を表す . さらに、この重心モーメントを次の式によって正規化し、正規重心モーメント η を求める .

$$\eta_{ij} = \frac{\mu_{ij}}{\mu_{00}^\gamma} \quad (5)$$

$$\gamma = \frac{i+j}{2} + 1, \text{ただし } i+j \geq 2$$

正規化することにより、分散の広がり具合がモーメント値に影響しなくなる (相似な分散であれば、同値になる) ため、図形のスケールに対して不変となる .

Hu モーメント不変量は、正規重心モーメントを組み合わせた式で、7 種類定義されているが最も簡単なのは以下の式である .

$$\varphi = \eta_{20} + \eta_{02} \quad (6)$$

これは、X 軸方向への分散と Y 軸方向への分散を足した値である .

3.2.3 3 次元 Hu モーメント不変量

提案手法では、Hu モーメント不変量を 3 次元に拡張して用いる . 座標 (x, y, z) におけるボクセル値を $vox(x, y, z)$ とすると、3 次元画像モーメントは以下の式で求められる .

$$m_{pqr} = \sum_{x,y,z} x^p y^q z^r vox(x, y, z) \quad (7)$$

これは、2 次元の場合と同様、原点 $(0, 0, 0)$ を中心としたボクセルの分散値であり、添字の p, q, r は、それぞれ X 軸方向、Y 軸方向、Z 軸方向に対する重みを表している .

続いて重心 $(\bar{x}, \bar{y}, \bar{z})$ を、以下の式によって求める .

$$(\bar{x}, \bar{y}, \bar{z}) = \left(\frac{m_{100}}{m_{000}}, \frac{m_{010}}{m_{000}}, \frac{m_{001}}{m_{000}} \right) \quad (8)$$

\bar{x} は X 軸方向の重心、 \bar{y} は Y 軸方向の重心、 \bar{z} は Z 軸方向の重心である .

この重心座標をもとにして、重心モーメント μ を以下の式で求める .

$$\mu_{pqr} = \sum_{x,y,z} (x - \bar{x})^p (y - \bar{y})^q (z - \bar{z})^r vox(x, y, z) \quad (9)$$

この値は、重心を考慮したボクセルの分散を表す . Hu モーメント不変量の式 ([10]) を 3 次元に拡張すると、以下のようになる .

$$\varphi = \mu_{200} + \mu_{020} + \mu_{002} \quad (10)$$

これが 3 次元 Hu モーメント不変量である . 2 次元の場合と同様に、これは X 軸方向への分散と Y 軸方向への分散と Z 軸方向への分散の和である .

3.3 回転検出と差分抽出

動き検出を用いることによってブロックの動きベクトルが求められるが、シーン予測を行うにはブロックの回転成分が必要である .

回転成分を求めるには、まず回転前のブロック (前フレームの動き補償範囲の中で最も不変量の近いブロック) と回転後のブロック (現フレームのブロック) における共分散行列を求める . ブロックの共分散行列 Σ は、空間内の重心モーメントより構成される .

$$\Sigma = \begin{pmatrix} \mu_{200} & \mu_{110} & \mu_{101} \\ \mu_{110} & \mu_{020} & \mu_{011} \\ \mu_{101} & \mu_{011} & \mu_{002} \end{pmatrix} \quad (11)$$

この共分散行列に対し、次の式で固有値分解を行う .

$$\Sigma = x \Lambda \quad (12)$$

共分散行列は対称行列であるため、 Λ は対角行列である . ここで $\lambda_1 > \lambda_2 > \lambda_3$ とする . このとき、固有ベクトル x の変化が回転量を表す .

すなわち、回転後のブロックの共分散行列を Σ' とし、これを固有値分解して得られたベクトルを x' 、また、この回転を表す回転行列を R とすると、以下の式が成り立つ .

$$x' = R \cdot x \quad (13)$$

この式より回転行列 R を求めることができる .

予測シーンは、前フレームのシーン、動きベクトル、および回転行列の 3 つの情報より求める . 前フレームのシーンの座標 (x, y, z) におけるボクセル値を $vox_{n-1}(x, y, z)$ 、動きベクトルを $V = (v_x, v_y, v_z)$ 、回転行列を R とすると、予測シーンの座標 (x, y, z) におけるボクセル値 $vox_p(x, y, z)$ は以下の式で求められる .

$$vox_p(x, y, z) = vox_{n-1}(x' - v_x, y' - v_y, z' - v_z) \quad (14)$$

$$\text{ただし、} \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = R^{-1} \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

この演算を全ての座標 (x, y, z) で行うことにより、予測シーンを生成する . 続いて、この予測シーンと実際の現フレームのシーンを比較し、その差分を抽出する . 現フレームのシーンのボクセル値を $vox_n(x, y, z)$ とすると、座標 (x, y, z) における差分値 $dif(x, y, z)$ は以下の式で求められる .

$$dif(x, y, z) = vox_n(x, y, z) - vox_p(x, y, z) \quad (15)$$

4. 評価実験

4.1 3 次元 Hu モーメント不変量の正当性評価

空間内のオブジェクトの移動や回転に対して 3 次元 Hu モーメント不変量が不変であるかどうかを調べた .

$32 \times 32 \times 32$ のボクセル空間において、図 3 に示す球 (半径 10)、直方体 ($11 \times 13 \times 15$)、立方体 ($11 \times 11 \times 11$) のそれ

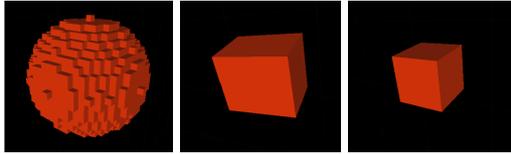


図 3 物体 (左:球, 中:直方体, 右:立方体)

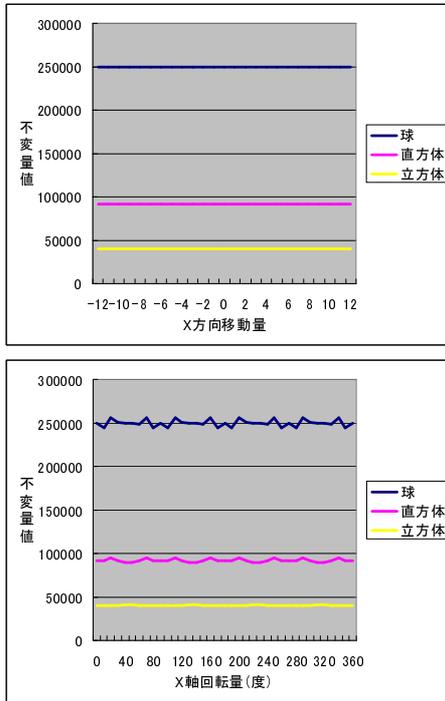


図 4 物体の状態変化による Hu 不変量値の推移 (上:平行移動, 下:回転)

それぞれの物体を置き, これに対して平行および回転移動を行った.

それぞれの物体をボクセル空間内で X 軸方向に平行移動させたところ, 図 4 (上) に示すように, 不変量は全く変化しなかった. また, 同様に Y 軸方向・Z 軸方向に平行移動させた場合も値は全く変化しなかった. 同様に, X 軸に平行で (16, 16, 16) を通る直線を回転軸として回転を行ったところ, 不変量値は図 4 (下) のように変化した. また, 回転軸を Y 軸, Z 軸に変えて回転させても全く同じ結果であった.

Hu 不変量が任意の平行移動に対して不変であったことから, 物体の平行移動に対しては不変性が非常に高いことが分かる. 物体の回転の場合は, ある程度の幅で値の変動が起きる. 初期状態の数値からの変化率が最大になる回転量は形状によって異なるが, その変化率はいずれも小さく, 3 次元 Hu モーメント不変量が物体の回転に対する不変量として十分使用できることが分かる.

4.2 3次元 Hu モーメント不変量の動き検出精度の評価

3次元 Hu モーメント不変量が, 3次元空間内で物体の動きを追うことができるかどうかを調べた.

ここでは, Poser で生成した人体の 3 次元 CG データをボリュウムデータに変換して, 2 フレーム間でのブロックマッチングのシミュレーションを行った. ここでは, 図 5 のように, 人体の直立シーンと, 人体を Y 軸周りに回転させたシーンを



図 5 Poser シミュレーションデータモデル (左:シーン 1, 右:シーン 2)

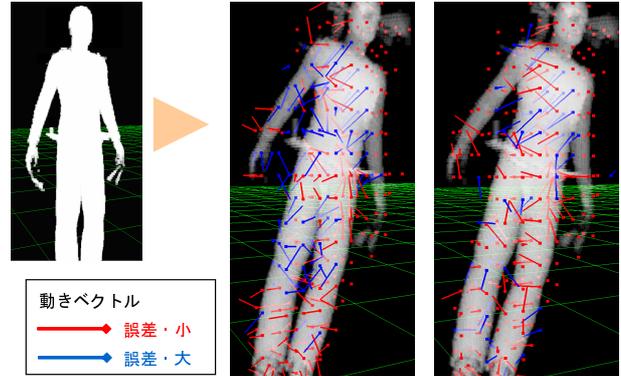


図 6 SAD と Hu 不変量で検出した動きベクトルの比較 (左: SAD, 右: Hu)

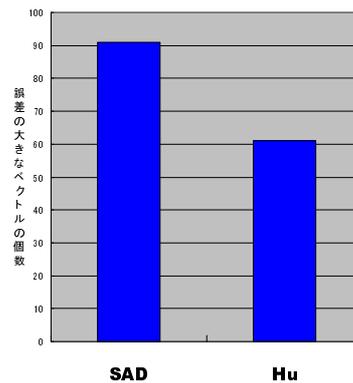


図 7 SAD と Hu 不変量で検出した誤差の大きなベクトルの個数

意した. この 2 つの CG データを元にボリュウムデータを生成した. この 2 つのデータが連続シーンであると仮定して, SAD を用いた動き検出と, Hu モーメント不変量を用いた動き検出の 2 種類の動き検出を行った.

この結果検出された各ブロックの動きベクトルを線分で視覚化したものを図 6 に示す. 赤いベクトルは動きベクトルの真値との差 (ベクトルの差の大きさ) がある閾値以下であるベクトルであり, 青いベクトルはその閾値を超えているベクトルである. この図を見ると, SAD で検出された動きベクトルは方向も一定でなく, 真値から外れたベクトルも多く見られるが, Hu 不変量で検出された動きベクトルは, 概ね回転と同じ方向を向いており, 真値から外れたベクトルも SAD と比べ少ない. 図 7 は各手法での検出の結果生じた, 誤差の大きいベクトル (図 6 の青いベクトル) の個数を示したグラフである. Hu では外

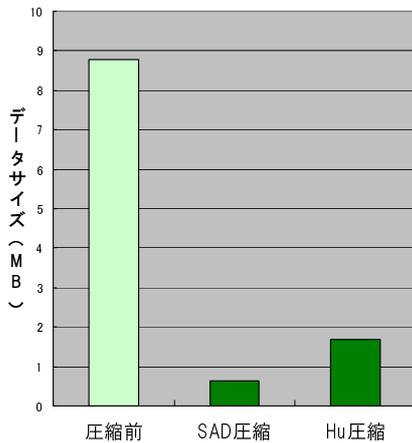


図 8 SAD, Hu 各手法における圧縮効率の比較

れたベクトルを SAD の約 3 分の 2 まで削減できていることが分かる。Hu 不変量を用いた手法は、SAD を用いた手法と比べるとより高い精度で回転運動を検出できると考えられる。

4.3 時系列ボリュームデータの圧縮効率の評価

本手法を用いて、実際にキャプチャリングスタジオで取得した人体の時系列 2 値ボリュームデータの時系列圧縮を行い、その圧縮効率を評価した。ここでは 44 フレーム (20fps) の 2 値時系列ボリュームデータを使用した。このデータの第 2 フレーム目以降の全フレームに対して、SAD 比較と Hu 不変量比較の 2 種類の手法で時系列ブロックマッチングを行った。

図 8 では、圧縮前のデータ、SAD 圧縮データ、Hu 圧縮データのファイル全体のデータサイズを示している。SAD は元データの 7.2% まで、Hu は 19.3% まで圧縮されており、圧縮効率では SAD が Hu を上回っていることが分かる。これに関しては、回転情報の分だけ情報量が多い、Hu 不変量が近いことと物体の類似性が高いことが必要十分な関係ではない可能性がある、動き補償空間のサイズが小さく、物体の運動を追跡しきれていないなど、いくつかの理由が考えられる。

5. おわりに

本稿では、3 次元 Hu モーメント不変量を用いた時系列ボリュームデータの圧縮法を提案した。時系列ボリュームデータにおいては、時間的に連続するフレームのデータは類似度が高いため、あるフレームのシーンを利用して次フレームの予測シーンを作成することができ、この予測との差分を抽出することでデータ量の削減を行った。予測シーンを作成するために、時系列ブロックマッチングによって類似ブロックを検索する必要があるが、この検索の際に 3 次元 Hu モーメント不変量を使用することで、物体の回転運動の検出を試みた。

実際に評価実験を行ったところ、3 次元 Hu モーメント不変量は、3 次元空間内でオブジェクトの平行移動や回転を行ってもわずかな変化しか起こさず、同一形状の判別には十分使用できると考えられる。また、オブジェクトの動き検出では、従来手法である SAD による動き検出に比べ、回転運動を正確に検

出できることが確認できた。しかし、圧縮効率に影響する数値である差分ボクセルの総和は SAD 手法よりも増加することが分かったため、今後、手法のさらなる改善が必要である。

今後の課題としては、SAD 手法と Hu 手法を融合させることによる圧縮効率の向上、また、3 次元 Hu モーメント不変量を用いた空間的圧縮法の提案などが挙げられる。

文 献

- [1] R.F. Rashid : " Towards a system for the interpretation of moving light display, "IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.2, No.6, pp.574-581, 1980.
- [2] S.Moezzi, L.Tai, P.Gerard: " Virtual view generation for 3D digital video, "IEEE Multimedia, pp.18-26, 1997.
- [3] G.Cheung, T.Kaneda: " A Real Time System for Robust 3D Voxel Reconrtruction of Human Motions, "in Proc. IEEE Conference on Computer Vision and Pattern Recognition, pp.714-720, 2000.
- [4] X.Gu, S.J.Gortler, H.Hoppe : " Geometry Images, "In Proceedings of the 29th annual conference on Computer graphics and interactive techniques, pages 355-361. ACM Press, 2002.
- [5] 森下広之, 大野義夫 : " MPEG アルゴリズムを適用した時系列ボリュームデータの圧縮 ", 情報処理学会研究報告-グラフィックスと CAD, Vol.1997, No.19, pp.33-38, 1997.
- [6] 韓昇龍, 山崎俊彦, 相澤清晴 : " 拡張ブロックマッチングによる 3 次元ビデオの圧縮に関する検討 ", 電子情報通信学会画像工学研究会技術報告, IE-20, 2005.
- [7] 東京大学大学院新領域創成科学研究科基盤情報学専攻相澤研究室 Web(<http://www.hal.k.u-tokyo.ac.jp/ja/research/3D.html>)
- [8] 波部斉, 曾良洋介, 松山隆司 : " Skin-off: テクスチャ指向型 2 次元平面展開による 3 次元ビデオの圧縮 ", 第 1 回デジタルシンポジウム講演予稿集, S1-2, 2005.
- [9] 安田浩, 渡辺裕 : " デジタル画像圧縮の基礎 ", 日経 BP 出版センター, 1996.
- [10] Hu M.K. : " Visual Pattern Recognition by Moment Invariants, "IRE Trans. Information Theory, Vol. 8, pp. 179-187, 1962.