

# 動き情報を考慮した動画像分散エンコーダの簡易圧縮

中根 翔† 高橋 浩平† 三浦 康之† 渡辺 重佳†

湘南工科大学†

## 1. はじめに

PC、デジタル家電の普及に伴い、PC が高性能化し、ソフトウェアによる動画像の編集や圧縮が可能になり、個人レベルで動画像を扱うことが多くなった。また、PC の普及に伴い、PC の低価格化により安価に並列処理が可能になっている。以上の背景から、動画像の圧縮にかかる時間を削減し、動画像の編集者の負担を減らすシステムの研究および開発を進めている<sup>[1][2]</sup>。それら一連の研究の一環として、LAN で伝送出来るようなデータの簡易圧縮法の検討を進めている<sup>[2]</sup>。現在検討中の手法は、単純差分の代わりに動き予測をすることで圧縮率を上げようとしている。

本稿では、単純差分の代わりに動き予測を使用することで簡易圧縮の向上を図る。

## 2. 並列ビデオエンコーダシステム

図1に、並列ビデオエンコーダシステムの構成を示す。本システムは、PC 端末から LAN を使いマルチコア PC に動画像を送って圧縮を行い、PC 端末に戻すものである。このシステムは通信網に特別線を使わずに、それぞれのクラスタ、利用者の PC を Gigabit Ethernet で結合している。利用者は、クラスタマシンに符号化する動画像を送り、クラスタマシンは、符号化処理が終了し次第、利用者の端末へ戻す。

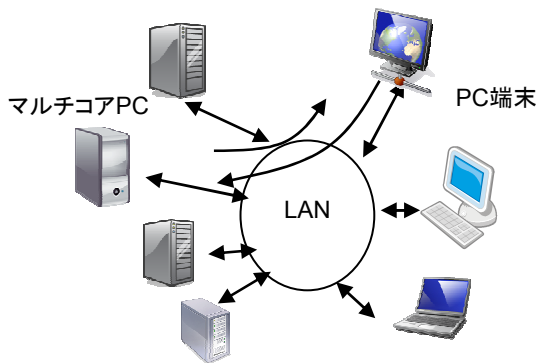


図1 並列ビデオエンコーダの図

## 3. 簡易圧縮アルゴリズム

図2に、簡易圧縮アルゴリズムの流れを示す。最初に今の画像と次の画像の2枚の画像をYU

V形式に変換し単純差分で画素の差を取り、その値を、RLE（三種類のRLE）を使いその後ハフマン符号化を行う。

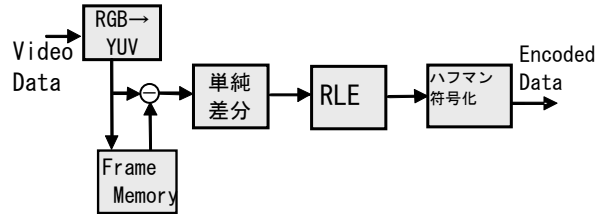


図2 符号化手順の流れ図

## 4 アルゴリズムの改善

本手法では、エンコーダで符号化を行う際に動きベクトルから生成されるため、生成された動きベクトルをユーザ端末にフィードバックすることにより、ユーザ端末側での簡易圧縮に利用する事が出来る。例えば、エンコーダで0フレーム目と1フレーム目のエンコードを行う際、動きベクトルが生成されるので、これをユーザ端末にフィードバックして、2フレーム以降の動きベクトルを予測する。

フレーム  $i$  とフレーム  $j$  の間の動きベクトル  $\vec{v}_{i-j}$  とすると、あらかじめ  $\vec{v}_{0-1}$  が与えられことになる。フレーム間の動きが等速であると仮定するした場合、

$$\vec{v}_{0-2} = 2\vec{v}_{0-1} \tag{1}$$

と予測することが出来る。同様に  $k$  フレーム目の動きベクトルは、以下のように予測することが出来る。

$$\vec{v}_{0-k,a} = k\vec{v}_{0-1} \tag{2}$$

ただし、ブロックマッチングを用いた動き情報は、物体の動きを正確に反映しない場合があるので、以下のようにするほうが適切な場合がある。

$$\vec{v}_{0-k,b} = \vec{v}_{0-1} \tag{3}$$

今回の手法では、フレームを  $16 \times 16$  のブロックに分割し、各ブロックについて、より誤差の少ない(圧縮後のデータ量の少ないことが予測される)結果を選ぶことで圧縮の効率の向上を図る。

## 5 実験

### 5.1 実験条件

An Improvement of Simple Compression Method with Motion Vector

† Sho Nakane, Kouhei Tkahasi, Yasuyuki Miura, Shigeyoshi Watanabe, Shonan Institute of Technology

映像情報メディア学会の標準動画をを用いて、連続したフレームに対して mobile multimerin system エンコーダから取り出したベクトルデータを使い単純差分と提案手法のデータ量を比較した。合計 30 種の画像を使用した。データを画像の大きさは 352×240 である。非圧縮の状態における画像のサイズは、257KB である。

5.2 実験結果

図 3 に、2 フレーム離れたフレームの動きベクトルを予測した場合の符号量を載せる。グラフから大きく改善できているもの、多少の改善が出来るもの、ほぼ改善の出来ないものがあることが見て取れる。図 3 のグラフから、改善が出来るものと出来ないものどでなぜ差が出たかを検討するため特徴のある画像のグラフをサンプルとしてそれぞれ載せる。図 4 に、画像全体が動いていく画像であり、図 5 に画像全体が無作為に小さく動く画像であり、図 6 に背景は動かないが一部が回転している画像であり、図 7 に背景は動かないが多くの人が無作為に動く画像の符号量を示す。

図 4 では提案手法である動き補償が狙い通り働いた結果大きく改善できているものであると考えられる。図 5 では動きが細かいせいか動き予測が効果的に働かなかったためにこのような結果になったと考えられる。図 6 では動き予測に誤情報が検出されていたためにデータ量が増えてしまっているものと考えられる。図 7 では図 5 に近い画像ではあったが一つ一つの動きが比較的大きいために多少の改善が出来るようである。

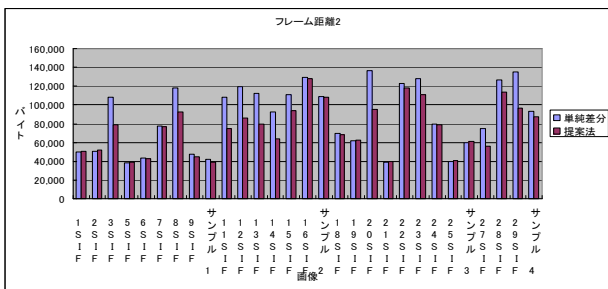


図 3 全体の符号量

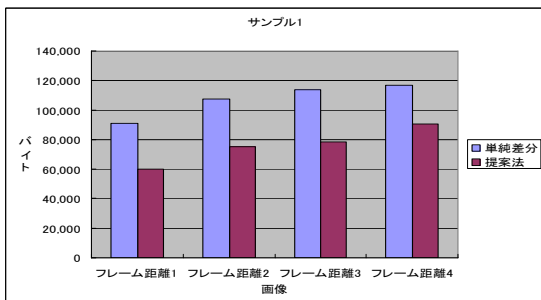


図 4 サンプル 1 の符号量

6 まとめ

本稿では、簡易圧縮のアルゴリズムの改善法を提案した。エンコーダから返ってきたデータには動きベクトルが含まれるため、それを利用した動き予測を行うことで差分を取った。30 種ほどの動画を用意し簡易圧縮後のデータを比較した。比較した結果からなぜそのような結果になったかを特徴のあるサンプルを取りだしそれぞれに対して検討した。等速で動く動画像では圧縮の効率化が図れたがそれ以外の画像では動き予測の特性上、予測自体がうまくいかない場合や、細かすぎて効果的に予測が行えないといった結果になり大きな改善は得られなかった。

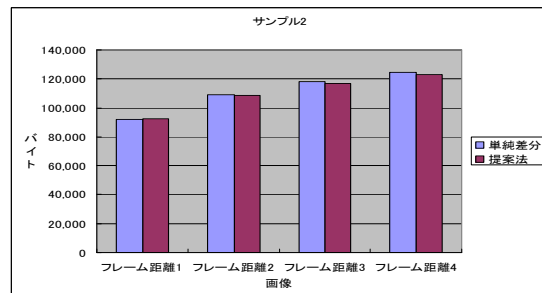


図 5 サンプル 2 の符号量

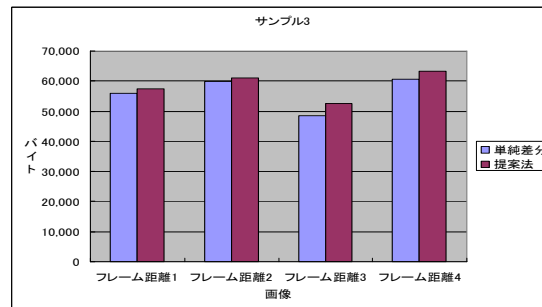


図 6 サンプル 3 の符号量

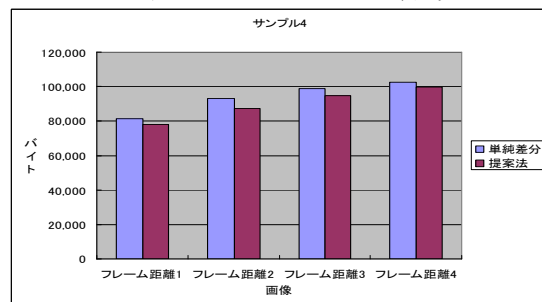


図 7 サンプル 4 の符号量

参考文献

[1] Yasuyuki Miura and Shogo Yamato, Simple Compression Method for Parallel Encoding Environment of Video Image, Proc. of the 2009 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing  
 [2] 中根翔、三浦康之、動画像並列エンコーダシステムにおける簡易圧縮の改善、第 72 回情報処理学会全国大会, 5ZD-3, 2010. 3.