

参照フレーム変更によるビットストリームスケールリング

清水 淳[†] 嵯峨田 淳[†] 渡辺 裕[†] 小林 直樹[†]

Bitstream Scaling by Reference Frame Altering

Atsushi SHIMIZU[†], Atsushi SAGATA[†], Hiroshi WATANABE[†], and Naoki KOBAYASHI[†]

あらまし インターネットでは、ユーザの接続環境により伝送レートや伝送経路の品質も様々である。このため、インターネット上で映像伝送サービスを提供するには、符号化ビットストリームから、実時間で任意の伝送レートに変換するビットストリームスケールリング技術が必要となる。従来、スケールリング方法として、DCT 係数の符号量を削減する方法や、他のフレームから参照されない符号化フレームを間引く方法が提案されている。しかし、インターネット映像伝送サービスを提供するには、従来方法のスケールリングの自由度では、不十分である。そこで、本論文では、演算コストを抑えた参照フレームを間引くビットストリームスケールリング方法を提案する。提案方法では、動きベクトル探索の代わりに、もとの動きベクトルから新たな動きベクトルを推定し、動き補償予測の参照関係を変更する。これにより演算コストを抑えつつ、スケールリングの自由度が拡大できることを示す。また、推定した動きベクトルと復号フレームから検出した動きベクトルを比較し、動きベクトル推定方法の確度についても検証する。

キーワード ビットストリームスケールリング, トランスコーダ, 再符号化, 動きベクトル推定

1. ま え が き

近年、インターネットによる映像配信サービスが広がっている。インターネット上で映像配信サービスを提供する場合、次の 2 点を考慮しなければならない。第 1 に、ユーザの接続環境により、利用可能な伝送レートが 10 数 kbit/s ~ 数百 kbit/s と多岐にわたる点である。第 2 に、インターネットは、混雑の割合により利用可能な帯域が変化する QOS (Quality of Service) 非保証のネットワークである点である。

これらの点から、インターネット映像配信サービスでは、ユーザからの要求やネットワークの状況に応じて、伝送レートが異なる複数のビットストリームを用意しなければならない。そこで、符号化したビットストリームから、任意の伝送レートに変換するビットストリームスケールリング技術が必要となる。更に、現在のインターネット映像配信サービスでは、その多くの機能がパソコン上のソフトウェアとして実現されていることから、パソコン上での実時間処理が望まれる。

ビットストリームスケールリング技術は、もとのビッ

トストリームの伝送を制限する方法と再符号化による方法に分類できる。

もとのビットストリームの伝送を制限する方法には、階層符号化方式の階層構造を利用する方法 [1] や DCT 係数などの直交変換係数の伝送を制限する方法 [1] が提案されている。階層符号化方式は、時間解像度や空間解像度などを階層化した符号化方式で、その伝送する階層数を変化させることでスケールリングができる。ただし、スケールリング可能な伝送レートの範囲 (以下、スケールリング自由度) を拡大するために階層数を多くした場合、もとのビットストリームの符号化効率が低下する問題がある。また、解像度が高い階層の符号化効率を重視して、高解像度の復号フレームを動き補償予測の参照フレームとした場合、低解像度の階層で、符号化器の局所復号フレームと復号器の復号フレームの不一致 (以下、ドリフト) が生じ、著しく画質が劣化する。この問題を解決するために、ドリフトの発生を抑制する方法 [2] やドリフトを抑制するための情報を伝送する方法 [3] が提案されている。また、階層構造をもたない符号化方式のビットストリームをスケールリングする方法として、直交変換係数の伝送を制限する方法がある。ただし、この方法は、伝送する直交変換係数を制限するため、ドリフトが生じ、スケール

[†] NTT サイバースペース研究所, 横須賀市
NTT Cyber Space Laboratories, Yokosuka-shi, 239-0847
Japan

グ後の画質劣化が顕著になる。

これに対し、再符号化による方法は、階層構造をもたない符号化方式のビットストリームもスケーリング可能であり、ドリフトも発生しないなどの特徴がある。また、もとのビットストリームの復号フレームを再符号化するため、スケーリング自由度が大きい。しかし、ビットストリームの伝送を制限する方法に比べ、スケーリング処理に要する演算量は増加する。そこで、再符号化方法の演算量を削減するため、もとのビットストリームの動きベクトルなどの情報を利用して再符号化を行う方法が提案されている [4], [5]。これらの方法は、動きベクトルの再探索が不要なため、演算量を抑えることができる。

100 kbit/s 程度の低ビットレートでは、DCT 係数の符号量が、全符号量の半分以下になることも珍しくない。このため、スケーリング自由度が DCT 係数の符号量に依存する再量子化方法では、インターネット映像配信サービスに要求されるスケーリングを実現できない可能性がある。そこで、スケーリング自由度を拡大する手法として、MPEG-1 [6] や MPEG-2 の双方向予測符号化フレームのように、他の符号化フレームから参照されない符号化フレームの伝送を制限することで、再符号化時の符号化フレームレートを下げる方法がある。しかし、他の符号化フレームから参照されない符号化フレームをもたない H.261 [7] や H.263 [8] などのビットストリームをスケーリングする場合、フレーム間予測の参照関係を変更しなければならない。符号化フレームの参照関係変更には、動きベクトルの再検出が必要であり、膨大な演算量を必要とするため、パソコンでの実時間処理が困難になる。

本論文では、既存の映像配信システムへの適用を考慮し、再符号化方法によるビットストリームスケーリング技術を提案する。提案方法では、もとの符号化フレームレートより再符号化フレームレートを下げることで、再量子化方法より、スケーリング自由度を拡大する。また、もとのビットストリームの動きベクトル情報を利用して、動きベクトルの再検出を行わずに再符号化フレームの参照関係を変更 [9], [10] し、実時間スケーリングを実現する。

以下、まず 2. で、従来の再符号化方法によるスケーリング方法について述べる。次に 3. で、参照関係変更によるスケーリング方法を提案し、その具体的な構成について述べる。続いて 4. で、符号化実験により、提案方法の有効性を示す。

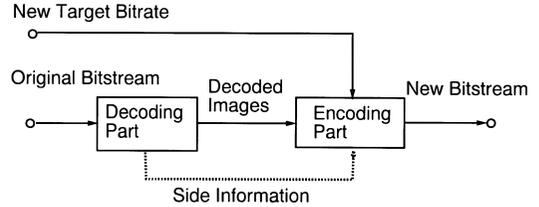


図 1 再符号化方法の基本構成

Fig. 1 The block diagram of re-encoding method.

2. 再符号化によるビットストリームスケーリング

再符号化方法は、復号部 (Decoding part) と符号化部 (Encoding part) から構成される (図 1)。復号部で、もとのビットストリームを復号し、符号化部で、復号フレームを再符号化ビットレートで符号化することによりスケーリングを行う。再符号化方法は、階層構造をもたない符号化方式のビットストリームもスケーリング可能であり、ドリフトも発生しないなどの特徴がある。しかし、ビットストリームを制限する方法に比べ、演算量が増加する問題がある。特に、符号化部が動き補償を用いる符号化方式の場合、動き探索の演算量が大きくなり、ソフトウェアによる実時間スケーリングは困難になる。

そこで、再符号化の際、もとのビットストリームに含まれる動きベクトル (Motion Vector) などの情報 (Side Information) を再利用することで、演算量を削減する方法が提案されている [4], [5]。この方法は、フレーム間予測の参照関係を変更せずに再符号化し、もとのビットストリームの動きベクトルや、フレーム内符号化/フレーム間符号化を示す符号化モード (Coding Mode) などの情報を再利用する。つまり、もとのビットストリームの DCT 係数を粗く量子化することで発生符号量を削減し、スケーリングを行う (以下、再量子化方法)。再量子化方法の構成を図 2 に示す。復号部では、もとのビットストリームから、動きベクトルや符号化モードなどを抽出する。符号化部では、復号部で抽出した動きベクトルや符号化モードをもとに新たな目標符号化ビットレート (New Target Bitrate) で符号化を行う。符号化部の構成は、通常の符号化器から符号化モード選択部と動きベクトル探索部を除いた構成となる。再量子化方法ではフレーム間予測の参照関係を変更しないため、符号化ビットレート制御部

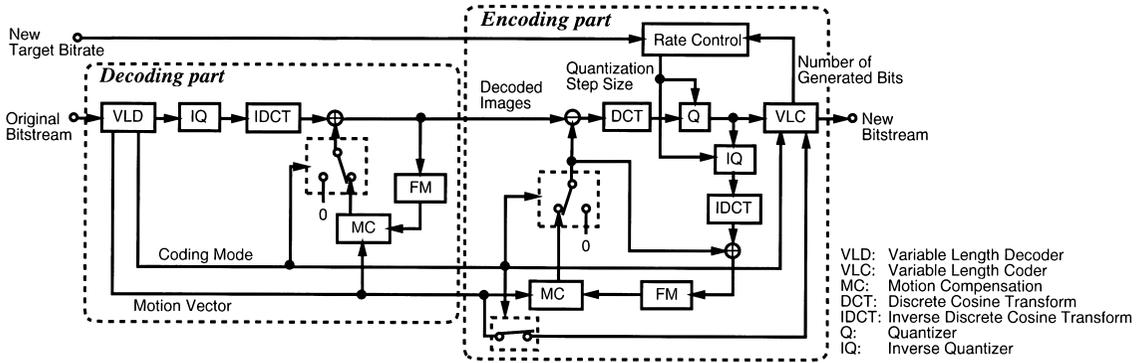


図2 再量子化方法の構成
Fig. 2 The block diagram of re-quantization method.

(Rate Control)では符号化フレームレートの制御は行わない。

このように、再量子化方法は、動き探索や符号化モード選択などの処理を省くことで、少ない演算量でスケールリングを実現できる。また、もとのビットストリームの符号化方式(階層構造や双方向予測フレームの有無など)による制約が少ないため、既存の映像配信システム等に利用できる利点がある。しかし、DCT係数の再量子化によりスケールリングを行っているため、削減可能な符号量は、もとのビットストリームのDCT係数の符号量によって決まる。つまり、再量子化方法のスケールリング自由度は、もとのビットストリームのDCT係数の符号量に依存していることになる。例えば、100 kbit/sのビットストリームを30 kbit/sに再量子化方法でスケールリングする場合、DCT係数の符号量の占める割合が70%以下のビットストリームでは、スケールリング不可能である。

3. 参照関係変更によるビットストリームスケールリング

3.1 フレーム間予測符号化の参照関係変更

再量子化方法のスケールリング自由度は、もとのビットストリームのDCT係数の符号量に依存している。その再量子化方法のスケールリング自由度を拡大する手段として、もとの符号化フレームを間引いて再符号化フレームレートを下げる方法がある。ただし、間引く符号化フレームが他の符号化フレームから参照されている場合、動き補償予測の参照関係の変更が生じ、動きベクトルの再検出が必要になる。符号化フレームの参照関係変更の例を図3に示す。図3において、符

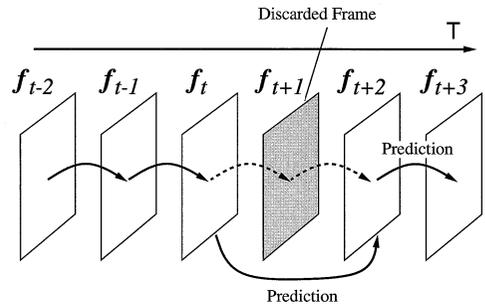


図3 参照フレームの変更
Fig. 3 Reference frame altering.

号化フレーム f_{t+1} は、符号化フレーム f_{t+2} の参照フレーム(Reference Frame)である。再符号化時に、符号化フレーム f_{t+1} を間引く場合、符号化フレーム f_{t+2} の参照フレームを符号化フレーム f_t に変更しなければならない。それには、符号化フレーム f_t から符号化フレーム f_{t+2} への動きベクトルの再探索が必要となり、演算量が増加する。したがって、従来の方法では、映像配信システムの符号化方式を変更するか、または、演算量増加に対応した高速な計算機を用意しなければならない。

そこで、動きベクトルの再探索をすることなく、もとのビットストリームに含まれている動きベクトルから、新たな参照関係の動きベクトルを推定し、参照関係を変更する方法を提案する。この推定方法により、間引く符号化フレーム(f_{t+1})の動きベクトルと、そのフレームを参照しているフレーム(f_{t+2})の動きベクトルから、新たな参照関係(f_t から f_{t+2} へ)の動きベクトルを推定する。

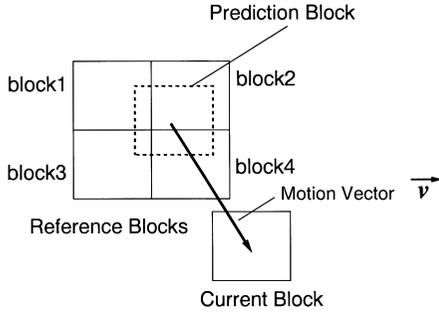


図4 動き補償
Fig. 4 Motion compensation.

3.2 動きベクトルの推定方法

H.263 などの動き補償予測符号化方式では、画像信号をブロック単位に分割して、符号化処理を行っている。動き補償予測では、各ブロックの動きベクトルにより参照フレーム (Reference Frame) の画像信号を平行移動させて、予測ブロック (Prediction Block) を生成する。符号化対象ブロック (Current Block) と参照ブロック (Reference Blocks), 予測ブロックの関係を図 4 に示す。ここで、参照ブロックは、参照フレームに含まれるブロックで、予測ブロックと重なるブロックである。

提案方法では、再符号化対象ブロックの動きベクトルと参照ブロックの動きベクトルから、新たな参照関係での動きベクトルを推定する。図 5 に、再符号化対象ブロック (Current Block) の動きベクトル推定に用いるブロック (もとの予測ブロック (Original Prediction Block), 参照ブロック (Original Reference Block), 新たな予測ブロック (New Prediction Block)) と各ブロックの動きベクトルの関係を示す。新たな動きベクトル \vec{v}_{new} は、新たな予測ブロックからもとの予測ブロックへの動きベクトル $\vec{v}_{new'}$ と、再符号化対象ブロックのもとの動きベクトル \vec{v}_{org} の和から求める。ここで、新たな予測ブロックからもとの予測ブロックへの動きベクトル $\vec{v}_{new'}$ は、もとの予測ブロックと重なる参照ブロック (block1~4) の動きベクトル \vec{v}_i ($i = 1, 2, 3, 4$) について、各参照ブロックともとの予測ブロックが重なる面積に応じて重み付けして算出するベクトルである。

以下に、動きベクトル推定方法の具体的な手順を説明する。動きベクトル推定の手順を図 6 に示す。まず、フレーム間符号化モード (Intra coded mode) の参

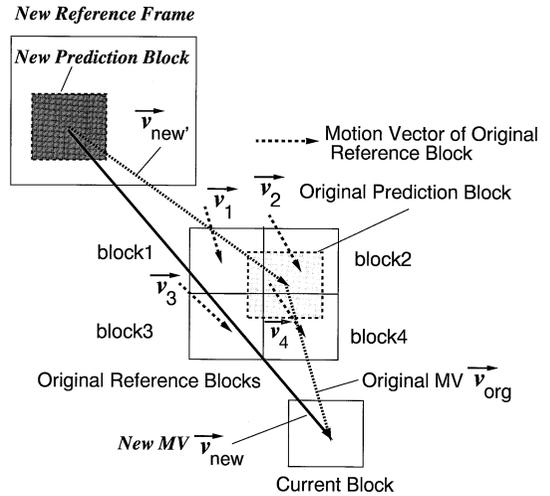


図5 動きベクトル推定の例
Fig. 5 An example of new motion vector estimation.

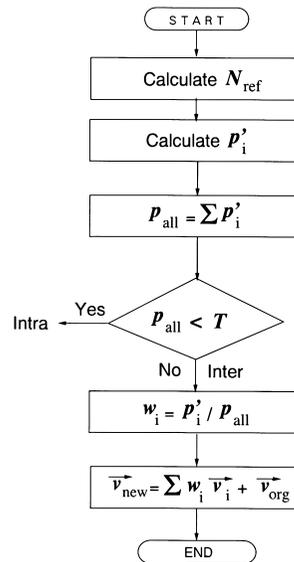


図6 動きベクトル推定の手順
Fig. 6 The flowchart of motion vector estimation.

照ブロックと予測ブロックが重なる部分の面積和 p_{all} を算出する。フレーム内符号化モード (Intra coded mode) の参照ブロックでは、もとのビットストリームに動きベクトルが存在しないため、予測ブロックと重なる部分の面積を 0 とする。

$$p_{all} = \sum_{i=1}^{N_{ref}} p'_i \quad (1)$$

$$p'_i = \begin{cases} p_i & \text{Inter coded mode} \\ 0 & \text{Intra coded mode} \end{cases}$$

ここで、 p_i は各参照ブロックで予測ブロックと重なる部分の面積、 N_{ref} は参照ブロック数で、予測ブロックの位置により 1~4 まで変化する。

このとき、フレーム間符号化モードの参照ブロックと重なる予測ブロックの面積 p_{all} がしきい値 T 以下の場合、再符号化対象ブロックの符号化モードをフレーム内符号化モードとして、動きベクトル算出を行わない。これは、フレーム内符号化モードの参照ブロックが、予測ブロックに占める比率が高い場合、正確な動きベクトルが得られないためである。

次に、参照ブロックの動きベクトル \vec{v}_i に対する重み係数 w_i を、各参照ブロックの占有面積 p'_i から算出する。

$$w_i = p'_i / p_{all} \quad (2)$$

続いて、新たな予測ブロックからもとの予測ブロックへの動きベクトル $\vec{v}_{new'}$ を次式より算出する。

$$\vec{v}_{new'} = \sum_{i=1}^{N_{ref}} (w_i \cdot \vec{v}_i), \quad \left(\sum_{i=1}^{N_{ref}} w_i = 1 \right) \quad (3)$$

新たな予測ブロックからもとの予測ブロックへの動きベクトル $\vec{v}_{new'}$ に、再符号化対象ブロックのもとの動きベクトル \vec{v}_{org} を加算して、新たな動きベクトル \vec{v}_{new} を求める。

$$\vec{v}_{new} = \vec{v}_{new'} + \vec{v}_{org} \quad (4)$$

上記の処理は、符号化フレームを 1 フレーム間引く場合である。複数のフレームを間引く場合、新たに求めた動きベクトル \vec{v}_{new} を $\vec{v}_{org} = \vec{v}_{new}$ として、上記の処理を繰り返し行う。例として、もとの符号化フレーム f_{t-1} と f_{t-2} を間引き、符号化フレーム f_{t-3} から f_t への動きベクトルを算出する場合を述べる (図 7)。

まず、符号化フレーム f_{t-2} から f_t への動きベクトル $\vec{v}_{t-2,t}$ を算出する。

$$\vec{v}_{t-2,t} = \sum_{i=1}^{N_{ref}} (w_i \cdot \vec{v}_i) + \vec{v}_{org} \quad (5)$$

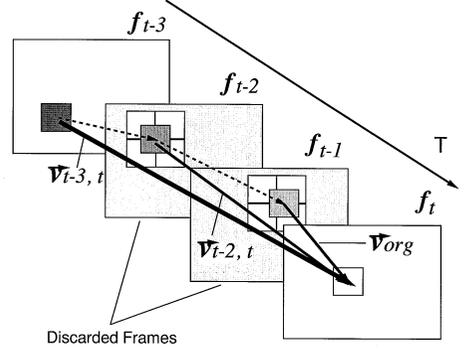


図 7 動きベクトル推定 (2 フレームを間引く場合)
Fig. 7 The estimation of new motion vector. (discarding 2 frames)

続いて、符号化フレーム f_{t-2} から f_t への動きベクトル $\vec{v}_{t-2,t}$ を用いて、符号化フレーム f_{t-3} から f_t への動きベクトル $\vec{v}_{t-3,t}$ を算出する。

$$\vec{v}_{t-3,t} = \sum_{i=1}^{N_{ref}} (w'_i \cdot \vec{v}'_i) + \vec{v}_{t-2,t} \quad (6)$$

このような処理を行うことで、参照関係変更後の動きベクトルを推定する。

3.3 システム構成

次に、提案方法の構成について説明する。提案方法は、五つの部分 (復号部、再符号化フレームレート算出部、動きベクトル推定部、符号化モード選択部、再符号化部) から構成される (図 8)。以下に詳細を述べる。

復号部では、もとのビットストリームを復号するだけでなく、もとの符号化ビットレート、符号化フレームレート、動きベクトル、符号化モードなどの情報を抽出する。

符号化フレームレート算出部では、再符号化時の符号化フレームレート F_{new} を算出する。再符号化時の符号化フレームレートの算出に用いるパラメータは、以下のとおりである。

- もとの符号化ビットレート R_{org}
- もとの符号化フレームレート F_{org}
- 再符号化ビットレート R_{new}
- 発生符号量 G_t

これらのパラメータから再符号化フレームレートを次式より算出する。

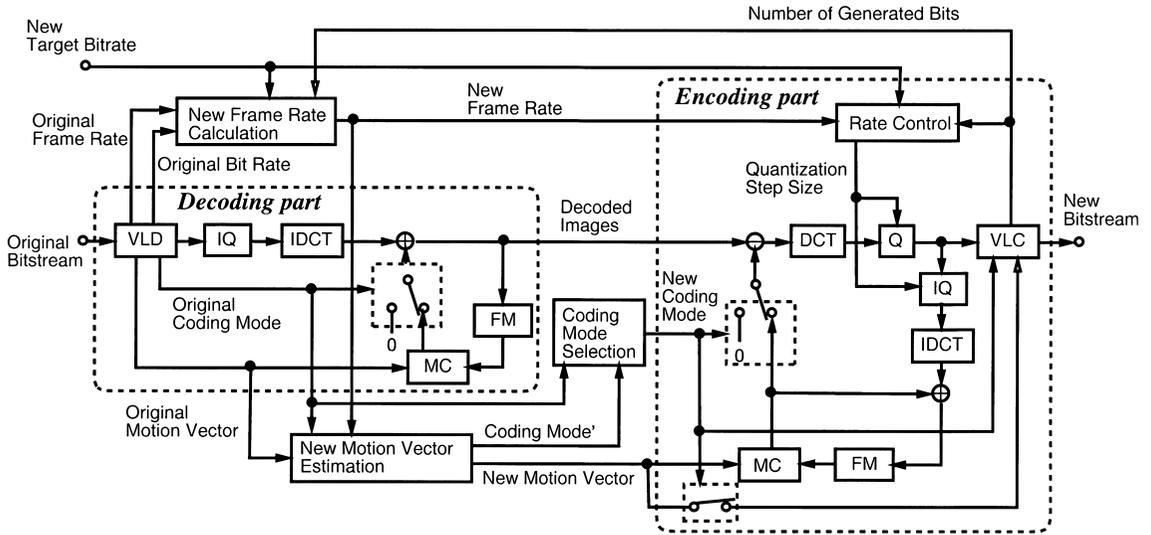


図8 提案方法のブロック図
 Fig. 8 The block diagram of the proposed method.

$$F_{new} = \alpha \cdot F_{org} \cdot \frac{R_{new}}{R_{org}} \quad (7)$$

ここで、 α は再符号化ビットレートを保証するための補正係数で、再符号化ビットレート R_{new} と再符号化時の発生符号量 G_t から算出する。この補正係数 α は、再符号化ビットレート R_{new} に比べ、再符号化時の発生符号量 G_t が多い場合は 1 以下の値となり、逆に、再符号化時の発生符号量 G_t が少ない場合は 1 以上の値となるように決定する。算出した再符号化フレームレートは、再符号化処理部の量子化制御と動きベクトル推定で利用される。

動きベクトル推定部では、再符号化フレームレートから間引くフレームを特定し、参照関係変更のための動きベクトルを推定する。

符号化モード選択部では、もとの符号化モード及び動きベクトル推定時に決定した符号化モードから、再符号化対象ブロックの符号化モードを決定する。両符号化モードともフレーム間符号化モードの場合、再符号化対象ブロックの符号化モードをフレーム間符号化モードとする。

再符号化処理部では、新たに求めた再符号化フレームレート、動きベクトル、符号化モードを用いて、再符号化ビットレートに収まるように復号フレームの符号化を行う。

表1 もとのビットストリームの仕様
 Table 1 The original bitstream spec.

Image Sequence	Coast Guard, Foreman (300 frames)
Image Size	176 [pel] × 144 [pel] (QCIF format)
Bit rate	112 [kbit/s]
Frame rate	15 [fps]
MC	Forward Prediction
ME	Full Search
ME Search Area	±15.5 [pel] × ±15.5 [pel]

4. 実験

4.1 実験条件

符号化アルゴリズムとして、動き補償予測 + DCT を用いた。簡単のために動き補償予測は順方向予測のみとする。もとのビットストリームは、2種類の10秒分の画像シーケンスを、符号化ビットレート 112 kbit/s で符号化したものを用いた。詳細を表1に示す。本実験では、再符号化ビットレートを 56 kbit/s, 28 kbit/s, 14 kbit/s の3通りとし、以下の3種類のスケールング方法について検討を行った。

- 再量子化方法 (Re-quantization)
- 再符号化方法 + 動き検出 (Re-encoding)
- 提案方法 (Proposed)

再量子化方法では、動きベクトルと符号化モードは、もとのビットストリームから抽出したものを用いた。再符号化方法+動き検出 (以下、再符号化方法) は、

表 2 再符号化フレームレート [fps]
Table 2 Coded frame rate. [fps]

Method	Target Bitrate		
	56 kbit/s	28 kbit/s	14 kbit/s
Re-quantization	15	15	15
Re-encoding	7.5	3.75	1.875
Proposed	7.5	3.75	1.875

もとのビットストリーム生成に用いた符号化器を使って、もとのビットストリームを復号した復号フレームを再符号化した。その際、再符号化時の動きベクトルは、 $\pm 15.5 \text{ pel} \times \pm 15.5 \text{ pel}$ の探索範囲で、全探索アルゴリズムにより検出した。

再符号化方法及び提案方法の符号化フレームレートは、符号化対象フレームを一致させて符号化効率を比較するため、式 (7) の補正係数 α を 1 として固定フレームレートとした。再符号化ビットレートと再符号化フレームレートの関係を表 2 に示す。

次に、提案方法で用いた動きベクトル推定方法で推定した動きベクトルの確度を検証するため、動きベクトルの確度の指標 (推定誤差) を定義する。推定誤差は、復号フレームに対し全探索で検出した再符号化方法の動きベクトル $\vec{v}_f = (x_f, y_f)$ と、もとのビットストリームに含まれる動きベクトルより推定した動きベクトル $\vec{v}_p = (x_p, y_p)$ の各成分ごとの差分絶対値 $A_{bs,x}, A_{bs,y}$ を用いる。

$$A_{bs,x} = |x_p - x_f|, \quad A_{bs,y} = |y_p - y_f| \quad (8)$$

また、スケールリングの演算量の目安として、スケールリングに要する処理時間を計測した。処理時間の計測は、400 MHz の CPU を搭載したパソコン上で行った。

4.2 実験結果と考察

スケールリング実験では、もとの符号化ビットレート 112 kbit/s から、56 kbit/s、28 kbit/s、14 kbit/s にスケールリングしたときの再符号化性能、動きベクトル推定方法、処理時間について検討した。

まず、スケールリング自由度について検討する。目標再符号化ビットレートと実際の再符号化ビットレートの関係を表 3 (Coast Guard) 及び表 4 (Foreman) に示す。これらの結果から、再符号化方法及び提案方法では、14 kbit/s へのスケールリングが可能であることがわかる。しかし、再量子化方法は、目標再符号化ビットレートを 14 kbit/s としてスケールリングを行ったにもかかわらず、再符号化後のビットレートは、Coast Guard で 21.06 kbit/s、Foreman で 22.29 kbit/s となってい

表 3 再符号化ビットレート [kbit/s] (Coast Guard)
Table 3 Re-encoded bitrate [kbit/s]. (Coast Guard)

Method	Target bitrate		
	56 kbit/s	28 kbit/s	14 kbit/s
Re-quantization	55.55	27.78	21.06
Re-encoding	55.22	28.00	14.00
Proposed	55.22	28.00	13.89

表 4 再符号化ビットレート [kbit/s] (Foreman)
Table 4 Re-encoded bitrate [kbit/s]. (Foreman)

Method	Target bitrate		
	56 kbit/s	28 kbit/s	14 kbit/s
Re-quantization	55.66	27.78	22.29
Re-encoding	55.22	28.00	13.88
Proposed	55.10	28.00	13.44

る。14 kbit/s にスケールリングした際の各方法の量子化パラメータの変化を図 9 に示す。量子化パラメータは、量子化ステップサイズの半分値であり、量子化制御部で算出されるパラメータである。横軸は再符号化フレーム番号、縦軸は量子化パラメータのフレームごとの平均値である。図 9 を見ると、どちらの画像も再量子化方法の量子化パラメータの値は、31 でほぼ一定となっている。この値は、量子化パラメータの最大値であり、DCT 係数の量子化制御だけでは、十分なスケールリングができないことがわかる。これに対して、再符号化方法及び提案方法では、目標再符号化ビットレートに合わせて、再符号化フレームレートを自由に変えられるため、14 kbit/s より更に低い再符号化ビットレートへのスケールリングを実現することが可能である。

次に、符号化効率について検討する。各方法の符号化効率比較のため、もとのビットストリームの復号画像と、再符号化ビットストリームの復号画像の SN 比を測定した。各方法について、復号画像の平均 SN 比を表 5 及び表 6 に示す^(注 1)。再量子化方法は、3 方法の中で最も SN 比が低くなっているが、これは、他の方法に比べ、再符号化フレームレートが高いため、1 フレーム当りの割当て符号量が少ないためである。提案方法の SN 比は、再符号化方法と比較すると、最大で 0.66 dB の低下に抑えられている。提案方法と再符号化方法の相違点は、動きベクトル検出方法だけであり、同一の復号フレームを再符号化対象フレームとしていることから、両方法の SN 比の差が、符号化効率

(注 1): 再量子化方法による 14 kbit/s へのスケールリングの SN 比は、再符号化ビットレートが 14 kbit/s を超えたため、除外した。

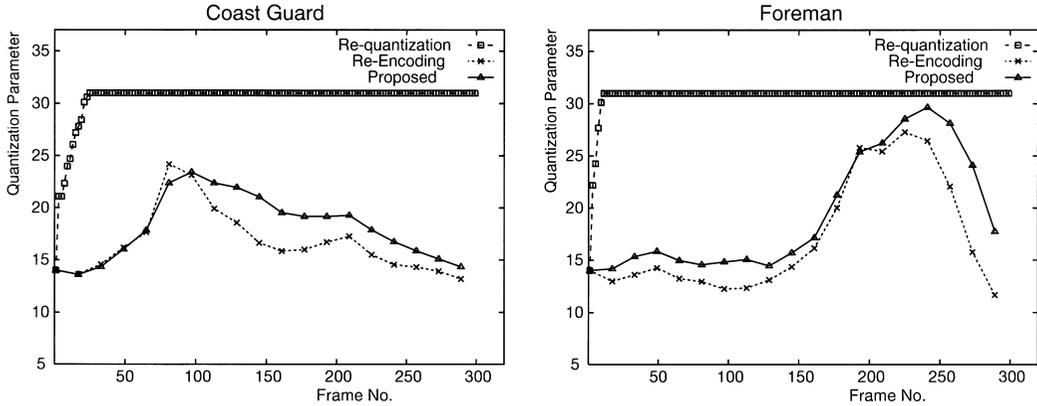


図9 14 kbit/s にスケーリングしたときの量子化パラメータ
 Fig. 9 The quantization parameter when scaling down to 14 kbit/s.

表5 PSNR [dB] (Coast Guard)
 Table 5 PSNR [dB]. (Coast Guard)

Method	Target bitrate		
	56 kbit/s	28 kbit/s	14 kbit/s
Re-quantization	30.82	27.67	—
Re-encoding	32.08	30.34	28.77
Proposed	32.06	30.15	28.27

表6 PSNR [dB] (Foreman)
 Table 6 PSNR [dB]. (Foreman)

Method	Target bitrate		
	56 kbit/s	28 kbit/s	14 kbit/s
Re-quantization	32.01	28.19	—
Re-encoding	32.86	31.02	29.83
Proposed	32.77	30.74	29.17

の差といえる．提案方法では，再符号化方法に対する SN 比の低下が小さいことから，再符号化方法と同程度の符号化効率であるといえる．

再符号化方法では，全探索アルゴリズムにより予測誤差が最小となる動きベクトルを検出している．このため，動きベクトルの推定誤差が大きくなるに従い，符号化フレーム間の予測誤差は増加する可能性がある．符号化フレーム間の予測誤差が増加すると，量子化後の有意係数が増えることで，発生符号量が増加し，目標符号化ビットレートを保証するためには，より大きい量子化ステップサイズで量子化しなければならない．つまり，符号化フレーム間の予測誤差の増加により，量子化ステップサイズが大きくなり，復号画像の SN 比が低下する．したがって，提案方法の SN 比低下の主な原因は，動きベクトルの推定誤差によるフ

レーム間の予測誤差の増加と考えられる．そこで，提案方法と再符号化方法の動きベクトルを比較し，動きベクトル推定による推定誤差について検討する．動きベクトルの推定誤差を図 10 (Coast Guard) と図 11 (Foreman) に示す．横軸は再符号化フレーム番号，縦軸は各成分の推定誤差のフレーム内平均値である．もとの符号化フレームを 1 フレーム間引いた場合，水平/垂直成分ともに，推定誤差は，ほぼ 1 画素以下になっている．また，3 フレーム間引いた場合でも，推定誤差は 3 画素程度に抑えられている．間引くフレームが多くなるに従って，推定誤差が大きくなる傾向が見られる．これは，間引くフレーム数に応じて，動きベクトル推定の回数が増え，誤差が蓄積しているためと推測される．また，Coast Guard (図 10) の 50 フレームから 100 フレームの間と Foreman (図 11) の 150 フレームから 250 フレームの間では，推定誤差が急激に大きくなっている．これは，動き量に対して探索範囲が狭いことが原因であり，探索範囲を拡大することで性能を改善できる．

これらの結果から，提案方法で用いた動きベクトル推定方法は，推定回数の増加に応じて，推定誤差が増加するものの，全探索で検出された動きベクトルに近い動きベクトルを推定可能であることがわかる．また，動きベクトルの推定誤差の増加に伴い，提案方法と再符号化方法の SN 比の差が拡大することから，推定誤差の増加が，符号化フレーム間の予測誤差の増加につながると考えられる．

次に，各方法の処理時間の比較を行う．各方法の処理時間を表 7 に示す．表 7 の計測結果は，400 MHz

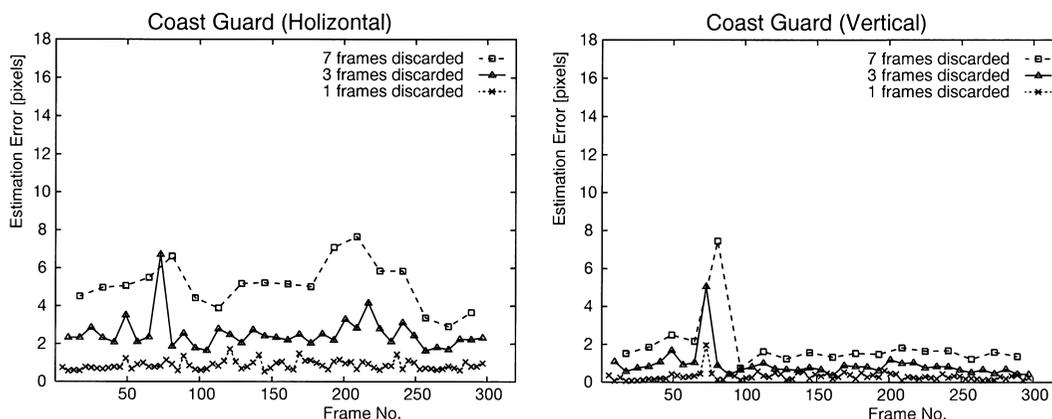


図 10 動きベクトルの推定誤差 (Coast Guard)
 Fig. 10 The estimation error of motion vector. (Coast Guard)

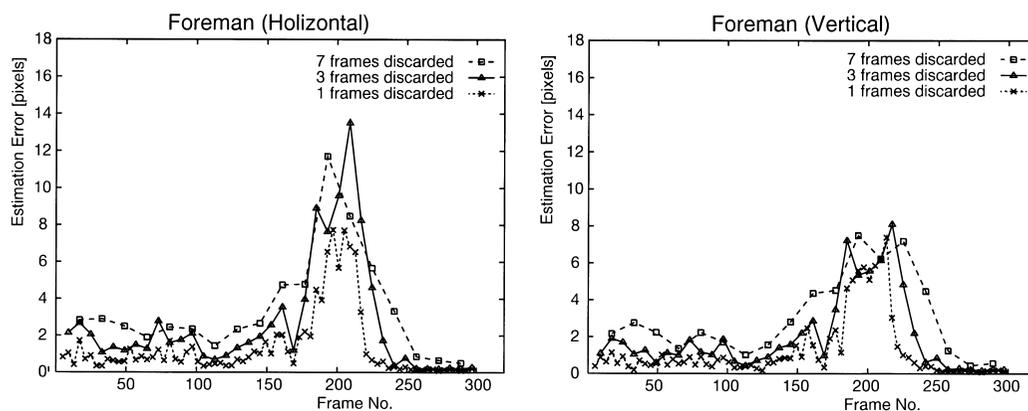


図 11 動きベクトルの推定誤差 (Foreman)
 Fig. 11 The estimation error of motion vector. (Foreman)

表 7 スケールリング処理時間 [s]

Table 7 Processing time of bitstream scaling [s].

Method	Target bitrate		
	56 kbit/s	28 kbit/s	14 kbit/s
Re-quantization	10.08	9.83	9.73
Re-encoding	69.78	39.99	21.90
Proposed	6.39	4.43	3.29

の CPU を搭載したパソコンで、10 秒分のビットストリームのスケールリングに要した時間である。この結果から、再量子化方法と提案方法では、ほぼ実時間で処理が可能であることがわかる。再量子化方法と提案方法のスケールリング処理時間を比較すると、提案方法のスケールリング処理時間が短く、再符号化ビットレートが小さくなるに従い、その差が広がっていることが

わかる。これは、再量子化方法では、再符号化フレームレートが一定であるのに対し、提案方法では、再符号化ビットレートが小さくなるに従い、再符号化部で処理するフレーム数が減少するためである。

これに対し、再符号化方法では、動き探索の処理量が多いため、実験に用いたパソコンでは、実時間でスケールリング処理が困難であることがわかる。高速動き探索アルゴリズムを用いることで、処理時間の短縮は可能ではあるが、符号化効率が低下する可能性がある。

これら実験結果から、提案方法は、再符号化方法と同程度の符号化性能で、スケールリング自由度が広く、スケールリング処理時間が短いという特徴があることがわかる。

5. む す び

本論文では、動き補償予測符号化において、符号化フレームの参照関係を変更するスケーリング方法を提案し、その有効性を実験結果により示した。

提案方法では、もとのビットストリームに含まれる動きベクトルから新たな動きベクトルを推定して、フレーム間予測の参照関係を変更している。これにより、少ない演算量で、スケーリング自由度をより拡大することができることを示した。また、もとのビットストリームに含まれる動きベクトルから推定した動きベクトルは、復号フレームに対して再探索を行った動きベクトルとの誤差が小さく、符号化効率の低下も抑えられることがわかった。

提案方法は、広いスケーリング自由度が要求されるインターネット映像伝送サービスに有効であるといえる。

謝辞 本研究の遂行にあたり、日ごろ御指導ご鞭撻いただく NTT サイバースペース研究所小暮賢司所長、一之瀬進プロジェクトマネージャ、並びにメディア通信プロジェクトの皆様へ感謝の意を表します。

文 献

- [1] ISO/IEC: "Information technology-generic coding of moving picture and associated audio," IS-13818, 1994.
- [2] O. Werner, "Drift analysis and drift reduction for multiresolution hybrid video coding," Signal Processing: Image Communication, vol.8, no.5, pp.387-409, July 1996.
- [3] R. Mathew and J.F. Arnold, "Layered coding using bitstream decomposition with drift correction," IEEE Trans. Circuits Syst. Video Technol., vol.7, no.6, pp.882-891, Dec. 1998.
- [4] H. Sun, W. Kwok, and J.W. Zdepski, "Architectures for MPEG compressed bitstream scaling," IEEE Trans. Circuits Syst. Video Technol., vol.6, no.2, pp.191-199, April 1996.
- [5] G. Keesman, R. Hellinghuizen, F. Hoeksema, and G. Heideman, "Transcoding of MPEG bitstream," Signal Processing: Image Communication, vol.8, no.6, pp.481-500, Aug. 1996.
- [6] ISO/IEC: "Coding of moving picture and associated audio for digital storage media up to about 1.5 Mbit/s," IS-11172, 1993.
- [7] ITU-T: "Video codec for audiovisual services at p x 64 kbit/s," Recommendation H.261, 1990.
- [8] ITU-T: "Video coding for low bitrate communication," Recommendation H.263, 1995.
- [9] 清水 淳, 嵯峨田 淳, 渡辺 裕, 小寺 博, "フレームスキップによるビットストリームスケーリング," PCSJ'96,

pp.25-26, 1996.

- [10] A. Shimizu, A. Sagata, H. Watanabe, and H. Kotera, "Bitstream scaling by reference frame altering," AVSPN '97, pp.183-187, Sept. 1997.

(平成 11 年 6 月 4 日受付, 9 月 27 日再受付)



清水 淳 (正員)

平 2 日大・理工・電子卒。平 4 同大大学院修士課程了。同年日本電信電話(株)入社。以来、低レート映像符号化方式の研究・開発に従事。現在、NTT サイバースペース研究所メディア通信プロジェクト研究主任。平 9 画像電子学会技術賞, 平 10 電子情報通信学会学術奨励賞各受賞。映像情報メディア学会, 画像電子学会各会員。



嵯峨田 淳 (正員)

平 6 東大・工・電子卒。同年日本電信電話(株)入社。現在 NTT サイバースペース研究所メディア通信プロジェクトにて低レート映像符号化方式の研究, スケーラブル符号化システムの研究開発に従事。映像情報メディア学会会員。



渡辺 裕 (正員)

昭 55 北大・工・電子卒。昭 57 同大大学院修士課程了。昭 60 同大大学院博士課程了。同年日本電信電話(株)入社。以来、画像の高効率符号化の研究を中心に、テレビ電話・テレビ会議用 CODEC の開発, 蓄積メディア用ビデオ符号化方式の研究, MPEG-1, 2 の標準化に従事。現在、NTT サイバースペース研究所メディア通信プロジェクト主幹研究員。昭 63 電子情報通信学会学術奨励賞, 平 9 画像電子学会技術賞, 平 9 映像情報メディア学会著述賞各受賞。工博。IEEE, 映像情報メディア学会, 情報処理学会, 画像電子学会, 日本音響学会各会員。



小林 直樹 (正員)

昭 54 東工大・理・応物卒。昭 56 同大大学院物理情報工学修士課程了。同年日本電信電話公社入社。以来、ファクシミリ, 画像入力, マルチメディアシステムの研究開発を経て, 現在サイバースペース研究所にて画像符号化の研究開発に従事。同研究所メディア通信プロジェクト画像符号化技術グループリーダー。TTC 第 5 部門第 2 専門委員長。