

MPEG-4 スプライト符号化を用いた超低レート動画像圧縮

秦泉寺久美^{†,††} 渡辺 裕^{††} 岡田 重樹[†] 小林 直樹[†]

Very Low Bit-Rate Video Compression Using MPEG-4 Sprite Coding

Kumi JINZENJI^{†,††}, Hiroshi WATANABE^{††}, Shigeki OKADA[†],
and Naoki KOBAYASHI[†]

あらまし 新しい画像符号化標準の MPEG-4 が標準化されつつある。これは、低いレートにおいては H.26X や MPEG-1, 2 に代表される従来符号化法と比較してより高品質の画像を提供するものである。また、新しい機能であるオブジェクト単位の符号化を提供するものである。筆者らはインターネット等に適用できる超低ビットレート符号化方法の開発を行っている。従来法に比べて劇的な符号量削減が可能である MPEG-4 の符号化ツールである「スプライト符号化」に着目し、前景、背景の 2 層からなるビデオオブジェクトの自動抽出アルゴリズムを提案する。生成した前景、背景のビデオオブジェクトをそれぞれ MPEG-4 オブジェクト符号化並びにスプライト符号化に適用（スプライトモード）して、スプライトを用いない MPEG-4 Simple profile の符号化法（ノーマルモード）と比較検討を行った。特に、フレームレートや前景オブジェクトの割合を変化させ、符号量に与える影響を調査した。その結果、フレームレートに関係なく、前景比率が画面全体の 10～15% 程度である場合に MPEG-4 通常符号化方法の 1/4～1/2 程度の符号量で同程度の画質を実現できることを確認した。また超低レート（128 kbit/s, 64 kbit/s）においてフレームレート、客観画質の評価を行ったところ、同程度の SNR で倍以上のフレームレートを達成できることを確認した。

キーワード MPEG-4, スプライト, オブジェクト符号化, 超高圧縮

1. ま え が き

筆者らはインターネットやモバイル通信環境に適用できる超低ビットレート符号化方法の研究開発を行っている [1], [2]。従来符号化法（H.26X 等）にて、超低レート（128 kbit/s 以下）等で画像を符号化すると、十分なフレームレートや画質を与えられないという問題があった。これは MC+DCT ベースのフレームごとの符号化方式の限界であり、これに代わる圧縮法が望まれていた。

本論文では、その実現方法の一つとして、新しい画像符号化標準 MPEG-4 [3] における「スプライト符号化」に着目し、動画像を前景、背景の 2 層のビデオオブジェクトに自動的に切り分け、それぞれを MPEG-4 オブジェクト符号化、MPEG-4 スプライト符号化する手法を提案する。

MPEG-4 では、オブジェクトベースの符号化を取り入れている。オブジェクト符号化では画像を意味あるオブジェクト単位に分割し、そのオブジェクトごとに符号化し、多重化して伝送する。利点として、オブジェクトごとにフレームレートや画質を変えることが可能である。スプライト符号化はオブジェクト符号化の一形態であり、スプライトを一つのオブジェクトとして扱うことが可能である。スプライト符号化は、カメラの動きのみを反映した背景画像に対して、カメラモーションに基づき画像を貼り合わせて 1 枚のパノラマ画像（背景スプライト）を作り、これを符号化に利用する^(注1)。大量のフレームが 1 枚の静止画像で表現でき、原画像との差分を一切符号化しないことから従来法に比べて劇的な符号量削減が期待できる [4], [5], [12], [13]。

しかし、スプライト符号化及びオブジェクト符号化の課題の一つとして、オブジェクトの自動生成が困難であるという点がある。例えば、動物体の抽出法にお

[†] NTT サイバースペース研究所, 横須賀市
NTT Cyber Space Labs., 1-1 Hikarinooka, Yokosuka-shi,
239-0847 Japan

^{††} 早稲田大学国際情報通信研究センター, 東京都
Waseda University, Global Information and Telecommunica-
tion Institute, Tokyo, 169-0051 Japan

(注1): 背景画像以外にも、オブジェクトの大局的な動きが一組の変換パラメータで表される場合はスプライトを構築することができる。背景スプライトはその一例である。

いては、半自動の抽出法 [6]、自動抽出法 [7] ~ [10] が提案されているが、オブジェクトが唯一存在する場合に限られる。多数のオブジェクトが存在する場合の異なるフレーム間での同一オブジェクトの対応問題が検討されされていないため、任意の画像に適用することが困難であった。

筆者らは文献 [11], [12] において、背景スプライトの自動生成法を提案し、生成された背景スプライト部分において、MPEG-4 スプライト符号化し、従来符号化法と比較して数倍の圧縮効率を達成している。しかし、前景部分の抽出並びに前景と背景を合わせた実際の符号化については未検討であった。本論文では、動画像を超低ビットレートでの符号化する方式の確立を目的とし、文献 [11], [12] の課題を克服するために、動画像から背景スプライトと前景オブジェクトを自動生成するアルゴリズムを提案する。また、提案アルゴリズムによって生成した前景、背景のビデオオブジェクトをそれぞれ MPEG-4 オブジェクト符号化並びにスプライト符号化して、スプライトを用いない MPEG-4 Simple profile の符号化 (ノーマルモード)^{注2)}との符号量の比較検討を行う。

はじめに、オブジェクトの対応問題を回避するために、前景オブジェクトと背景スプライトの 2 層からなる VOP (Video Object Plane) を定義する。VOP は従来符号化法のフレームに相当するもので、ビデオオブジェクトは一連の VOP の集合体である。2 層ビデオオブジェクト生成アルゴリズムは、スプライト生成に特化したグローバルモーション (GM) 算出法、高品質背景スプライト生成法、前景オブジェクト抽出法からなる。GM 算出法には筆者らが文献 [11], [12] で提案しているカメラモーション算出法を用いた。高品質スプライト生成においては、本論文では、従来の時間メジアン法 [4]、オーバーライト法^{注3)}の長所を生かした方法を新たに提案する。本手法では、前景のないかつ高品質なスプライト生成することが可能である。前景オブジェクト抽出法は、原画像と背景スプライトの差分を基本とし、GM のずれに対してロバストな方法を提案する。自動生成における前景オブジェクト形状は輪郭が複雑であること、孤立点が多数含まれることなどから予測がほとんどあたらず、その結果イントラ符号化されて符号量が増大する。この問題を回避するために形状をマクロブロック^{注4)}で近似する方法を提案する。更に、本論文では、提案アルゴリズムを用いて生成した 2 層ビデオオブジェクトを MPEG-4 スプライト

イト符号化し、VOP 構造をもたない MPEG-4 通常符号化方法と比較検討を行った。特にフレームレートや前景の大きさが符号量に与える影響について調査し、スプライト符号化が劇的な圧縮率を達成できる条件について考察を行った。更に、同条件下にて同一ビットレートでの符号化実験を行い、フレームレートや SNR を用いた客観画質の観点から比較検討を行った。

以降、2. では 2 層ビデオオブジェクト自動生成アルゴリズムを提案する。3. では MPEG-4 符号化実験について述べ、4. で結論を述べる。

2. 2 層ビデオオブジェクト自動生成アルゴリズム

本章では、2 層ビデオオブジェクト自動生成アルゴリズムについて述べる。本アルゴリズムでは、画像を前景と背景の二つのビデオオブジェクトに切り分け、背景ビデオオブジェクトを 1 枚のスプライトで表現する。多数の前景領域を一つの前景オブジェクトとして抽出することにより、多数の前景オブジェクトのフレーム間対応問題を回避できる。例えば、「サッカー」を想定すると、選手、審判員、ボールなどはすべて一つの前景オブジェクトとして扱える。また、背景スプライトを生成することで、スプライト符号化に特徴的な劇的な符号量削減を実現できる。

図 1 に 2 層ビデオオブジェクトを有する符号化復号モデルを示す。まず、符号化側で、画像を前景オブジェクトと背景スプライトの 2 層のビデオオブジェクトに分割する。背景スプライトはカメラの動きのみを反映した背景部分である。前景オブジェクトは背景以外のあらゆる動領域で、これらはすべて一つの前景として扱われる。前景オブジェクトと背景スプライトは独立なビデオオブジェクトとしてそれぞれ MPEG-4 オブジェクト符号化、MPEG-4 スプライト符号化 (スプライトモード)、多重化により一つのビットストリームとして送信する。受信側では、ビットストリームを多重分離し、ビデオオブジェクトごとに復号し、合成して、表示する。

図 2 に 2 層 VOP (Video Object Plane) 自動生成

(注 2): 以降、スプライトを使用する MPEG-4 符号化をスプライトモードでの符号化、スプライトやオブジェクトを使用しない従来方式を踏襲した MPEG-4 符号化をノーマルモードによる符号化と呼ぶことにする。

(注 3): 画像を上書きしながら貼り合わせてパノラマ画像をつくる一般的な手法。

(注 4): 16 画素四方で形成される、処理単位。

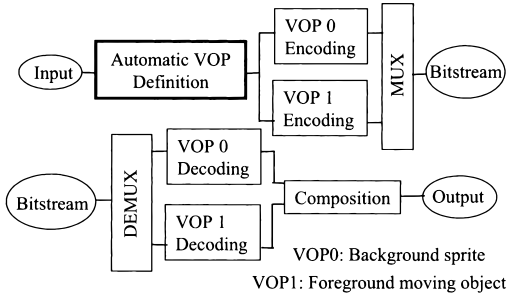


図1 MPEG-4符号化復号モデルと2層ビデオオブジェクト
 Fig.1 MPEG-4 codec model and two-layer video objects.

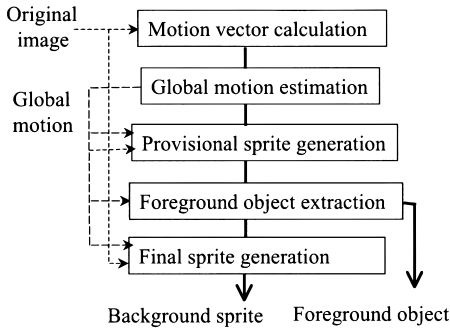


図2 2層ビデオオブジェクト自動生成アルゴリズムの概略
 Fig.2 Outline of two-layer video objects generation algorithm.

アルゴリズムの概略を示す。本アルゴリズムは主にGM算出、背景スプライト生成、前景オブジェクト抽出の三つの部分からなる。

まず、隣接フレーム間にて動きベクトルを算出する。次に、その動きベクトルを用いて基準フレームからのGMを算出する。このGMによって原画像を幾何変換して基準座標上にマップし、時間方向にメジアン(中央値)をとってスプライトの値として、動物体を消去した仮スプライトを生成する。更に、仮スプライトと原画像の差分画像から前景候補画像と背景画像を生成する。前景候補画像はマクロブロック近似されて最終的な前景画像となる。一方で、原画像から前景領域を除外された背景画像を幾何変換して基準座標上にマップして背景スプライトが生成される。以下にその詳細を述べる。

2.1 GM算出

スプライトを生成するためには、カメラ操作を反

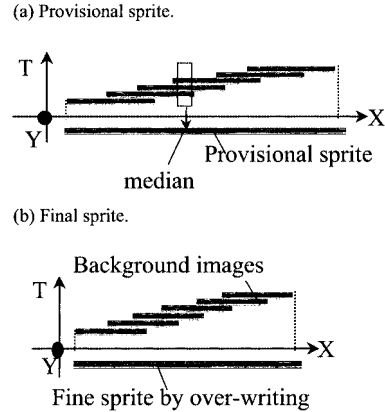


図3 XYT時空間における画像のアラインメントの様子
 Fig.3 Alignment of images in XYT spatio-temporal field.

映したGMを算出する必要がある。本論文では、文献[12]において筆者らの提案したGM算出法を用いる。本方法は以下の原理を利用している。

- (1) カメラモーションを反映した局所的動きベクトルにおいて、そのx方向(水平方向)並びにy方向(垂直方向)の変化量を軸とする特徴空間で独特の分布を示すことを利用
- (2) カメラモーションの候補となる複数のクラスターをそれぞれ検定し、最適なものをGMとして算出(クラスター検定)

算出された各々のフレーム間GMを用いて、基準座標からのGMにそれぞれ変換される。以降、GMとは基準座標からのGMを指す。

2.2 背景スプライト生成

前景のない、かつ、高品質なスプライトを生成するために、本論文では従来の時間メジアン法[4]、上書き法の長所を利用した方法を提案する。生成されるスプライトは仮スプライトと本スプライトの2種類である。

図3に画像が時間方向へ統合される様子を示す。スプライトとは画像を模様がつながるように並べてパノラマ画像にしたものである。前節で算出されたGMを用いて各々の原画像を幾何変形し(x, y, t)基準座標系の格子点上にマッピングする。基準座標系の任意の座標(x, y)に注目すると、時間(t)方向に複数の画素が重なっている。これら画素値のメジアンをとってその座標の値とする。ここで、動物体領域がその座標にマップされた画素数の半分より少ない場合を想定する。この場合は、座標上には背景を代表する画素が半数以

上を占めるので、背景を代表する画素がメジアンとして選択される。これをすべての座標で行くと動物体が消去されたスプライトが生成される。本論文では、このスプライトを仮スプライトと呼ぶ。しかし、時間メジアン法によって生成された仮スプライトは、動物体は消去されているものの、微小な GM のずれによって座標上に線形近似された値が候補として並んでいることから、スプライトが若干ぼけるという性質があった。そこで、この仮スプライトを前景抽出に利用する。

一方、上書き法は、画素をそのまま基準フレームに貼り付けていくので画質の良いスプライトが生成できるが、一番手前になるフレームの前景と、各フレームの端の前景オブジェクトが残るという問題があった。よって、動物体を抽出した後、動物体を取り去った部分を背景画像とし、それを基準座標に GM で幾何変換して順次上書きし、最終的な高画質スプライトを生成する。本論文では、これを本スプライトと呼ぶ。

2.3 前景オブジェクトの抽出

前景オブジェクト生成は、前景候補領域の抽出と前景形状のマクロブロック近似の 2 段階の手順を経て生成される。更に、前景比率によって抽出される前景を制御する。次項以降にその詳細を述べる。

2.3.1 前景候補領域の抽出

最初に、仮スプライトから GM によって切り出された背景画像と原画像を用いて、絶対差分画像 D を生成する。ここで、絶対差分画像 D を適当なしきい値 Th_0 で 2 値化し、2 値化画像 T を得る。差分の大きいところは動物体である可能性が高いので、この 2 値化画像 T を前景候補領域とする。

2.3.2 マクロブロック単位の前景オブジェクトの特定

自動生成における前景オブジェクト形状は輪郭が複雑であること、孤立点が多数含まれることなどからほとんど予測があたりずイントラ符号化され、その結果、符号量が増大する。本論文ではこの問題を回避するために、オブジェクト形状をマクロブロックで近似する方法を提案する。図 4 に前景形状のマクロブロック近似の様子を示す。

MPEG-4 において、オブジェクト形状符号化は可逆、非可逆の 2 通りのモードが用意されている [3], [13]。その粗さはパラメータ $alpha_{Th}$ で表される。一番粗い形状表現はマクロブロックで近似される。具体的には、マクロブロック内に前景情報が半数以上あればマクロブロック全体を前景、半数より少なければマクロ

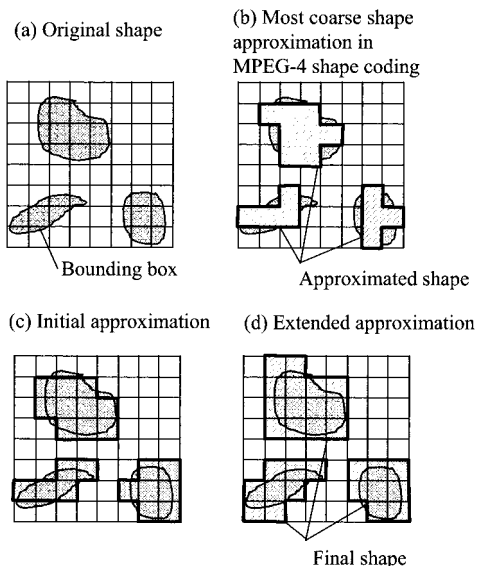


図 4 前景オブジェクト形状のマクロブロック近似の様子
Fig.4 Macro-block approximation for foreground object shape.

ブロック全体を背景とするものである [13]。もとの形状を図 4(a) としたとき、MPEG-4 における形状符号化の一番粗い近似法を図 4(b) に示す。この方法は前景オブジェクトがかなり侵食されて視覚的に妨害になるという問題があった。そこで、本論文では、新たなマクロブロック近似法を提案する。本手法は初期マクロブロック近似と拡張マクロブロック近似の二つの手法からなる。図 4(c) に初期マクロブロック近似、図 4(d) に拡張マクロブロック近似の様子を示す。マクロブロック内に前景がしきい値 Th_1 画素以上ある場合、マクロブロック全体を前景と判断する。それ以外のものは背景とする。次に、初期マクロブロック近似では前景となったマクロブロックに隣接する背景マクロブロックにおいて、そのマクロブロック内に前景画素がしきい値 Th_2 ($Th_2 < Th_1$) 画素以上あるものを前景と再判断する。拡張マクロブロック近似は N 回行う。このように、提案したマクロブロック近似法は、初期マクロブロック化によって前景オブジェクトの大まかな場所を特定する。更に拡張マクロブロック化によってその周辺の領域も前景オブジェクトに統合するという作用がある。提案手法でマクロブロック近似されたオブジェクト形状は MPEG-4 形状符号化における可逆モードで符号化される。

2.3.3 前景率固定下における前景オブジェクト抽出

最適なしきい値 Th_1, Th_2 は画像に依存する。例えば、複雑な背景をもつ画像では、GM が誤抽出された場合、差分による前景候補領域は必然的に多くなる。これをすべて前景領域と判定すると、前景の面積が大きくなり、符号量の増大につながる。一方で、一様なテクスチャを背景にもつような画像では、GM 誤抽出によって抽出される前景候補領域は少ないと考えられる。そこで、しきい値 Th_1, Th_2 を自動で決定するために、一定前景比率の前景オブジェクトを抽出するアルゴリズム (図 5) を提案する。基本的には、前述の初期マクロブロック化と拡張マクロブロック化の組合せを繰り返し行い、任意の前景比率 (前景オブジェクトの大きさの画像全体の大きさに対する割合) を処理の終了判定に使用する。

具体的なアルゴリズムの流れを以下に説明する。最初に、 V_{map} を初期化する。 V_{map} は、前景候補マクロブロックの場合は $V_{map}(i, j)$ ((i, j) はマクロブロック位置を示す) に 1 を格納し、背景マクロブロックは 0 を格納するものである。前述の 2 値化画像 T を用いて、各マクロブロック中の前景候補画素数をカウントしてマップ $C_{map}(i, j)$ を算出する。更に前景候補画素数の最小値 C_{min} を算出する。初期マクロブロック化のしきい値 Th_1 を C_{min} 、拡張マクロブロック化のしきい値 Th_2 を Th_1 より小さい任意の値に初期設定する。ここでは $Th_2 = Th_1/2$ を用いている。初期設定されたしきい値 Th_2 と Th_1 を用いてマクロブロック化を行う。ここで前景比率 ($V_{map}(i, j) = 1$ のマクロブロック数を総マクロブロック数 M で除した値) を計算し、任意の割合 Th_3 を下回らない場合、処理を続行する。このとき、マクロブロック化しきい値 Th_1 と Th_2 を $Th_1 = Th_1 + k_{step}, Th_2 = Th_1/2$ のように更新する。ここで、 k_{step} は任意の刻み幅である。次章に述べる実験では $k_{step} = 2$ を用いた。前景比率が任意の割合 Th_3 を下回ったとき、処理を終了する。処理を終了したときのマクロブロック化結果が前景オブジェクトとなる。

形状情報を有する標準画像 “Stefan” において、人物部分の形状情報をマクロブロック近似したところ、ロスレスで形状符号化する場合に比べて形状符号量が 1/10 程度に抑えられた。マクロブロック近似の他のメリットとして、ソフトウェアデコーダ実装時の処理コスト軽減が挙げられる。形状情報の復号、パディング

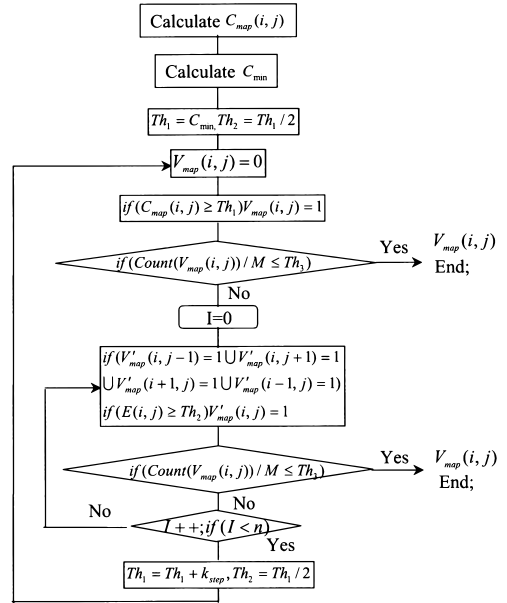


図 5 前景比率固定下における前景オブジェクト抽出アルゴリズム

Fig.5 Macro-block approximation algorithm under constant foreground ratio.

といった処理をしないのでリアルタイム復号が求められるアプリケーションに好都合と考えられる。

3. MPEG-4 符号化実験

スプライト符号化は、複数フレームにわたる背景部分を 1 枚の静止画で表し、原画像との差分を一切符号化しないことから、従来符号化に比べて劇的な圧縮率を達成可能であることが報告されている [4], [5], [11], [12]。しかし、画像によってはスプライトモードで符号化しても劇的な符号量削減ができない場合が考えられる。その要因としてフレームレートと前景比率が挙げられる。フレームレートが少ない場合、必然的に背景スプライトで表現できるフレーム数が少なくなり、劇的圧縮効果が得られない可能性がある。また、前景比率がある程度以上ではスプライトモードで符号化しても符号量削減にはならないことが予測される。これは、前景比率が 100% の場合 (画面全体が前景となった場合)、ノーマルモードに比べてスプライトモードはスプライトの分だけ情報量が増大することから明白である。ある程度の前景比率のところスプライトモードとノーマルモードの符号量が同じになると考えられる。3.1 及び 3.2 では、表 1 に示すカメラ操作を

表 1 MPEG-4 スプライト符号化実験に用いた画像
Table 1 Video sequences used for MPEG-4 sprite coding experiments.

Video	Type	Frames	Content
Horsrace	SIF	150	pan, two horses and a car
Soccer	SIF	150	pan and tilt, several players
Athlete	SIF	150	pan, a couple of runner
Stefan	SIF	150	pan and zoom, a player
Board	SIF	150	pan and tilt, a border

含む五つの画像において、提案アルゴリズムによって生成されたビデオオブジェクトをスプライトモードで符号化（図中では“Sprite”）し、スプライトを用いないノーマルモード（同、図中では“Rectangle”）と比較し、劇的な圧縮が可能である動画の条件を検討した。ここでは、量子化パラメータ QP を固定にして両モードの符号量を比較検討した。これは、文献 [12] における、人工的に作られた前景領域をもたない画像 (“MITsequence”) に関して、同程度の QP でノーマルモードとスプライトモード^(注5)で符号化した画像は同程度の主観画質を与えるという知見に基づく。本実験では“Stefan”を除いては背景がカメラに因る動きを除いてはほぼ静止しているものを用いた。よって、人工画像から得た知見を自然画像にほぼ適用できると考える。また、3.3 では、低レート時 (128 kbit/s, 64 kbit/s) にて両モードでの画質とフレームレートを客観的に評価した。

ここで、ノーマルモードとスプライトモードの前景オブジェクト部分の符号化において、GOV の取り方によって有利不利が生じないように、一つのシーケンスにつき一つの GOV 構造とした。すなわち、最初の I-VOP 以外はすべて P-VOP として符号化される。ノーマルモードにおいては一番符号量の小さくなる GOV 構造である。また両モードにおいて、MV 探索範囲は ± 32 画素とした。スプライトモードにおける GM はこの探索範囲で求められた MV を利用して算出した。

3.1 前景比率と符号量の関係

最初に、圧縮率に大きく寄与すると考えられる前景比率を 5~40% の範囲で可変にして両モードでの符号量測定を行った。画像サイズは SIF、フレームレートは 30 fps (frame per sec.)、量子化パラメータ QP は 12 で固定とした。また、スプライトモードの符号量は、前景オブジェクトと背景スプライトの符号量の合計を 1 秒当りの符号量に換算した値を用いた。また、各オブジェクトの符号量はそれぞれ、テキストチャ、モーショ

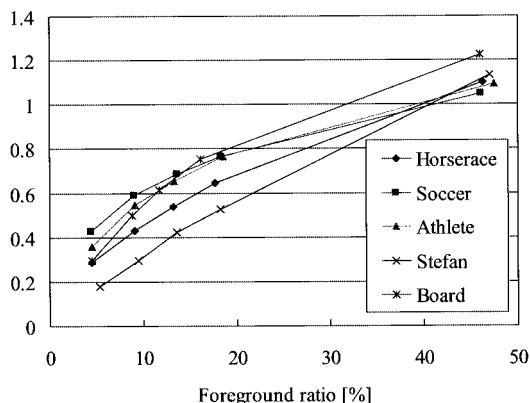


図 6 前景比率と MPEG-4 スプライト符号化における符号量の関係。MPEG-4 通常符号化時の符号量との比で表現

Fig. 6 Foreground ratio and coding efficiency under sprite mode coding condition. The ratio of coded bits to one of MPEG-4 normal coding are shown.

ン、形状の符号量の合計である。

図 6 に前景比率を変化させ、ノーマルモードとスプライトモードの符号量の比率を測定した結果を示す。横軸は前景比率、縦軸にスプライトモードの符号量をノーマルモードの符号量で除した値である。縦軸が値 1 を取るのは両符号化モードによる符号量が同じであることを示している。前景比率が 10~15% 以下の場合、スプライトモードがノーマルモードの半分以下の符号量を達成できた。また、前景率が 40% 程度でスプライトモードとノーマルモードの符号量が同程度になった。

3.2 フレームレートと符号量の関係

次に、任意の前景比率において、五つの画像のフレームレートを 5, 10, 15, 30 fps と可変にし、両モードでの符号量を測定した。ただしスプライトはフレームレート 30 fps の状態で作成したものを符号化に適用している。量子化パラメータは $QP = 12$ で固定とした。

図 7 に前景比率 10~15% 程度のときの五つの画像のスプライトモードとノーマルモードの符号量の関係を示す。スプライトモードがノーマルモードの 1/2 から 1/4 の符号量を達成していることがわかる。図 8 に $QP = 12$ で符号化時の“Horsrace”の両モードの

(注 5): 文献 [12] では各々 LME: Local Motion Estimation, SS: Static Sprite と表記。

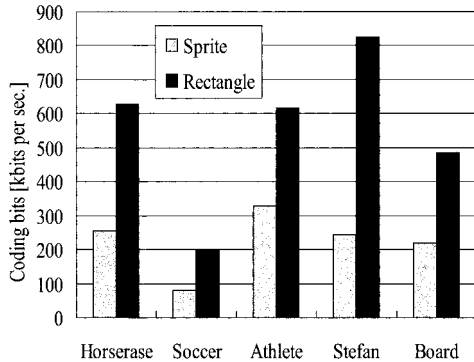


図7 前景比率 10～15%程度, 30 fps における MPEG-4 スプライト符号化と MPEG-4 通常符号化の符号量の比較

Fig. 7 Coding result of MPEG-4 sprite coding and normal coding under 10-15 percent of foreground ratio, 30 fps.

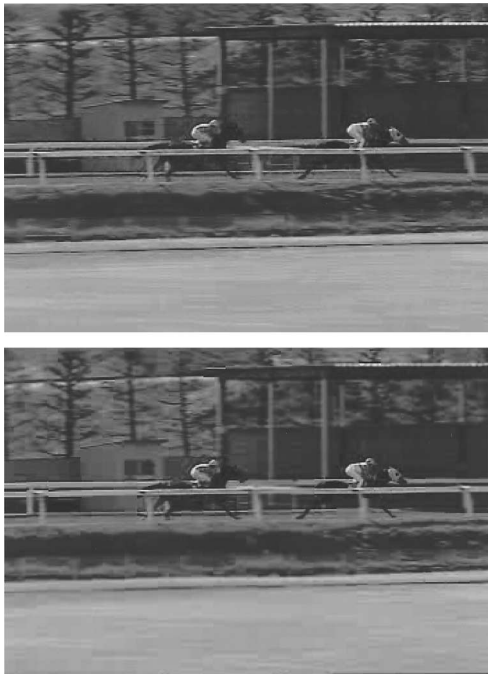


図8 $QP = 12$ 固定で符号化時の符号化画像の例 (“Horserace”). 上段: ノーマルモード, 下段: スプライトモード

Fig. 8 Examples of coded image under $QP = 12$ condition (“Horserace”). Upper: normal mode, lower: sprite mode.

符号化画像を示す．主観画質に関してはほとんど差がないことがわかる．ちなみに画像 “Horserace” のスプ

ライトモードにおける平均 SNR は 26.22 [dB], ノーマルモードにおける平均 SNR は 30.98 [dB] であった．SNR から主観画質が単純には比較できないことについては以降の 3.3 で述べる．

図 9 に本アルゴリズムによって生成された背景スプライトの例, 図 10 に原画像, 前景オブジェクト, 合成画像の例 (符号化画像) を示す．前景比率は 10～15% 程度である．前景の消去されたスプライトが生成されていることがわかる．また, 前景オブジェクトはマクロブロック単位に抽出されるので本来は背景に属する領域も前景オブジェクトとして抽出されている．しかし, 背景スプライトから背景を切り出し, 背景画像と前景を合成すると違和感のない画像が生成されていることがわかる．合成部を拡大してみると, $QP = 12$ で符号化時のブロックひずみと区別がつかないほど良好に合成されていることがわかる．

また, 図 11 にスプライトモード及びノーマルモードにおいて, フレームレートを変化させた場合の符号量を示す．前景比率は 10～15% 程度である．同程度の画質を達成するのに, スプライトモードはノーマルモードに比べて $1/2$ から $1/4$ の符号量で実現できている．この傾向は “Horserace” を除いてどのフレームレートにも当てはまる．“Horserace” は被写体やカメラの動きがかなり速いため, 低フレームレートではノーマルモードでは動きベクトル探索範囲が十分確保できず, 動き補償が効果的でなかったためと推定される．“Athlete” を例にとると, ノーマルモード 200 kbit/s 相当の符号量では 5 fps (frame per sec.) しか達成できないのに対し, スプライトモードでは 20 fps 以上を達成可能である．スプライトを使用すると, インターネットのような狭帯域環境下でもたくさんのフレームを表現でき, 低ビットレートでも動き滑らかな映像を SIF 等の大きな画像サイズで実現可能であることがわかる．

基本的にスプライトモードは “Stefan” のようにカメラが同じ場所を何度も往復して撮像する場合に最も威力を発揮する．この場合, シーケンスが長ければ長いほど, 1 秒当りのビット数は少なくなる．また, “Horserace” や “Athlete” のように一方方向にパンをしている画像は, スプライトの大きさはシーケンスの長さに比例して大きくなる．よって, 特にほぼ定速でパンしている場合などは 1 秒当りのビット数はどの長さにシーケンスを区切ろうとも一定の値になる．しかし, この場合でも劇的な符号量削減が可能であることが実



図 9 背景スプライトの抽出例
Fig.9 Examples of background sprite.

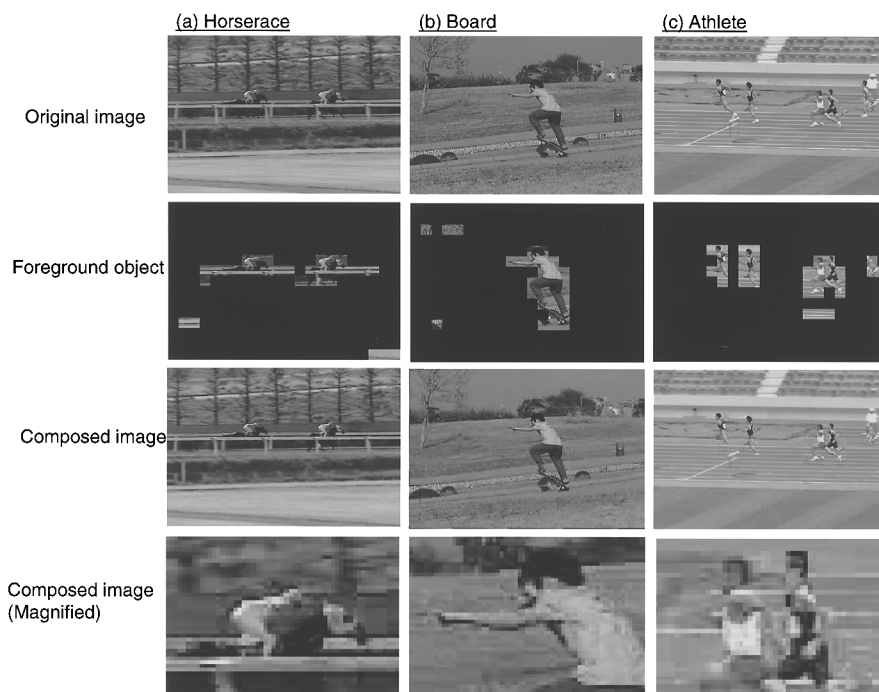


図 10 原画像, 前景オブジェクト, 合成画像の例
Fig.10 Examples of original image, foreground object and composed image.

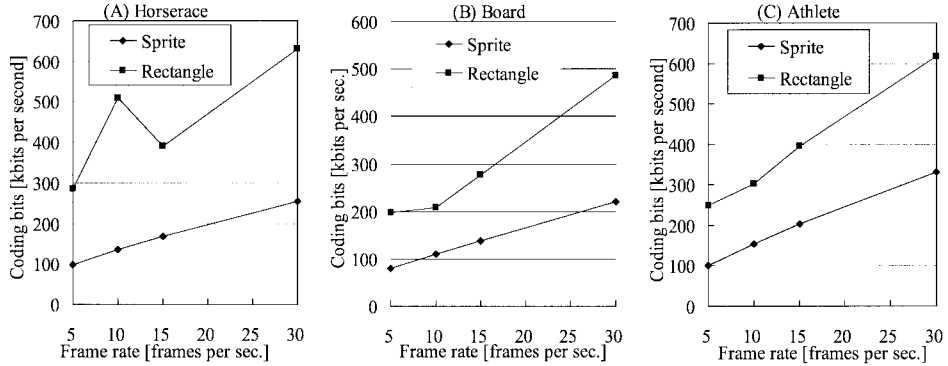


図 11 ノーマルモード及びスプライトモードにおけるフレームレートと符号量の関係
 Fig. 11 Framerate and coding efficiency under MPEG-4 sprite and normal coding condition.

験結果が示している。

ところで、画像“Stefan”は背景の観客が無視できないほど動いているので、本来前景として抽出されるべき領域が前景比率の制限のため抽出されず情報が欠落し、その結果情報量が削減された可能性が高い。その他の“Stefan”を除いた画像は、視覚的にはカメラ操作による動き以外の変化はほとんど検知できないが、1, 2画素程度微妙に動いているものもあると考えられる。これらの微小領域は制限前景比率内では抽出されず、符号化されていない。しかし、これら領域はごく微小であり、仮にこれらの領域を前景に回復したとしても前景比率には大方影響が出ないと考えられる。よって、劇的な符号量削減は情報の欠落がもたらしたのではないということが推測できる。

処理時間は、動きベクトル算出、画像処理（GM算出以降）、符号化全体の処理時間の合計を100%とすると、それぞれ78%、18%、4%であった。画像処理、符号化にかかる処理量は動きベクトル算出時間に比べて小さいことがわかる。

3.3 同一ビットレート符号化時のフレームレートと画質における評価

本節では実際に低レートでの符号化を想定し、上記の“Horserace”画像にて、ビットレート128 kbit/s、目標フレームレート15 fps、及び、ビットレート64 kbit/s、目標フレームレート10 fpsにてノーマルモードとスプライトモードで符号化実験を行った。ノーマルモードの符号化制御及びスプライトモードの前景オブジェクトの符号化制御はVM Ver.15 [13]に基づく。背景スプライトのフレームレートは前景オブジェクトのフ

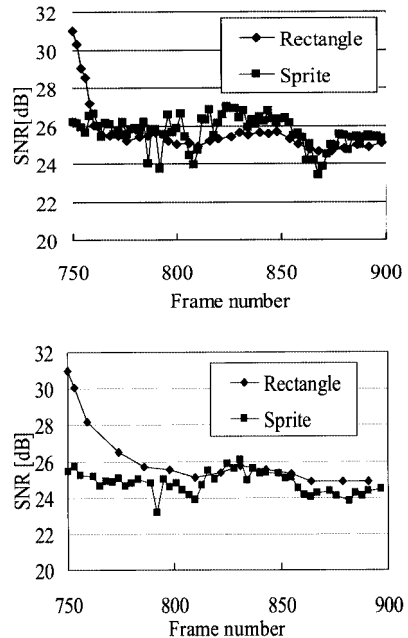


図 12 同一ビットレート符号化時の SNR による画質評価
 上段：128 kbit/s、下段：64 kbit/s
 Fig. 12 Objective coded image quality under same bitrate. Upper: 128 kbit/s and lower: 64 kbit/s.

レームレート制御に同期させている。スプライトモードの前景比率は前節の知見より、15%とした。評価はフレームレート並びに SNR による客観画質による。

また、図 12 に 128 kbit/s、64 kbit/s における各画像のフレームごとの SNR を比較した様子を示す。図 13

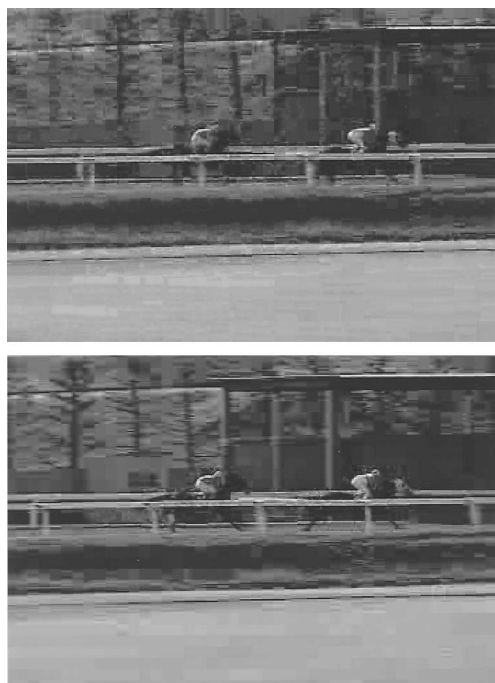


図 13 128 kbit/s におけるノーマルモード及びスプライトモードの符号化画像の例。上段：ノーマルモード，下段：スプライトモード

Fig. 13 Coded images under 128 kbit/s coding condition. Upper: normal mode, lower: sprite mode.

にノーマルモードとスプライトモードにおける符号化画像の例を示す。128 kbit/s において、ノーマルモードは平均 SNR が 25.85 [dB]、達成フレームレート 6.80 fps に対し、スプライトモードでは、平均 SNR が 25.69 dB、達成フレームレート 15 fps であった。同様、64 kbit/s においては、ノーマルモードの平均 SNR が 26.31 [dB]、達成フレームレートが 2.8 fps に対し、スプライトモードでは平均 SNR が 24.79 [dB]、達成フレームレート 8.60 fps であった。同一ビットレート下において、スプライトモードがノーマルモードの倍以上のフレームレートを達成できることが確認された。

更に、前節の図 8 の“Hoserace”のスプライトモードの合成画像と本節の図 13 の“Hoserace”の両モードの画像は明らかに前者の方が主観画質は良いが、SNR による客観画質は同程度である。このことから、SNR から単純に主観画質を述べることはできないことを記しておく。また、前節の実験では、前景比率が 10～15%程度の条件で、スプライトモードが同

程度の主観画質をノーマルモードの $1/2 \sim 1/4$ の符号量で達成できると報告した。本節では、同条件のもと、スプライトモードが同程度の符号量にて、同程度の主観画質、倍以上のフレームレートを達成したこと述べた。双方の実験結果はスプライトモードが前景比率がある程度小さいという条件のもと、ノーマルモードに比較して劇的な圧縮効率を得るという点で一致する。

4. む す び

インターネットやモバイル通信環境下における超低ビットレートの符号化方式の開発を目的として、本論文では、スプライト符号化の課題の一つであるビデオオブジェクトの自動抽出アルゴリズムを提案した。また、提案アルゴリズムを MPEG-4 スプライト符号化に適用して MPEG-4 通常符号化方法と比較、検討を行った。前景比率が画像全体の 10～15%程度に収まる条件下で、MPEG-4 通常符号化方法の $1/4 \sim 1/2$ 以下の符号量で同程度の画質を実現した。また、実際に超低レート (128 kbit/s, 64 kbit/s) で符号化実験し、同程度の SNR においてスプライトモードがノーマルモードの倍以上のフレームレートを達成できることがわかった。

このように、スプライト符号化は特定の条件のもとで、劇的な符号量削減が期待できる。しかし、すべてのビデオシーケンスがこの条件を満たしているわけではなく、スプライト符号化できるショットを判定する方法の検討が必要である。また、実際に配信を考慮した場合のビデオオブジェクトごと符号量割当ての検討が今後の研究課題である。

謝辞 本研究を進めるにあたって計算機シミュレーションに多大な協力を頂いた (株)NTT ソフトウェアの米原紀子氏に感謝致します。また、有益なディスカッションを賜った NTT サイバースペース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の皆様へ感謝致します。

文 献

- [1] K. Jinzenji, S. Okada, H. Watanabe, and N. Kobayashi, “Automatic two-layer video object plane generation scheme and its application to MPEG-4 video coding,” IEEE ISCAS2000, pp.606–609, May 2000.
- [2] 秦泉寺久美, 渡辺 裕, 小林直樹, “二層ビデオオブジェクト自動生成法と MEEG-4 符号化への適用,” 信学技報, IE99-115, Dec. 1999.
- [3] Final Draft of International Standard, ISO/IEC 14496-2.

- [4] M. Irani, S. Hsu, and P. Anandan, "Video compression using mosaic representation," *Signal Processing: Image Communication*, vol.7, pp.529-552, 1995.
- [5] M. Lee, W. Chen, C.B. Lin, C. Gu, T. Markoc, S.I. Zabinsky, and R. Szeliski, "A layered video object coding system using sprite and affine motion model," *IEEE Trans. Circuits & Systems for Video Technology*, vol.7, no.1, Feb. 1997.
- [6] M. Kass, A. Witikin, and D. Terzopoulos, "SNAKES: Active contour models," *Proc. 1st ICCV*, pp.259-268, 1987.
- [7] J.G. Choi, S. Lee, and S. Kim, "Spatio-temporal video segmentation using a joint similarity measure," *IEEE Trans. CSVT*, vol.7, no.2, pp.279-286, April 1997.
- [8] A. Neri, S. Colonnese, and G. Russo, "Automatic moving objects and background segmentation by means of higher order statistics," *IEEE ISCAS'97*, June 1997.
- [9] R. Mech and M. Wollborn, "A noise robust method for segmentation of moving objects in video sequence," *IEEE ICASSP'97*, April 1997.
- [10] T. Meier and K.N. Ngan, "Automatic segmentation of moving objects for video object plane generation," *IEEE Trans. CSVT*, vol.8, no.5, pp.525-538, Sept. 1998.
- [11] K. Jinzenji, H. Watanabe, and N. Kobayashi, "Global motion estimation for static sprite production and its application to video coding," *IEEE ISPACS'98*, pp.328-332, Nov. 1998.
- [12] 秦泉寺久美, 渡辺 裕, 小林直樹, "スプライト生成のためのグローバルモーション算出法と符号化への適用," *信学論 (D-II)*, vol.J83-D-II, no.2, pp.535-544, Feb. 2000.
- [13] "MPEG-4 Video Verification Model Version 15.0," ISO/IECJTC1/SC29/WG11 MPEG98/N3093.

(平成 12 年 7 月 3 日受付, 11 月 7 日再受付)



秦泉寺久美 (正員)

平 1 上智大・理工・電気電子卒。平 3 早大大学院修士課程了。同年 NTT に入社。高機能高効率通信のための画像処理, 符号化の研究に従事。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, 早稲田大学国際情報通信研究センター非常勤客員研究員(平 13 年 2 月~)。平 10 学術奨励賞受賞。IEEE, 映像情報メディア学会各会員。



渡辺 裕 (正員)

昭 55 北大・工・電子卒。昭 57 同大学院修士課程了。昭 60 同大学院博士課程了。同年 NTT 入社。NTT 研究所において, 画像の高効率符号化の研究, MPEG の標準化に従事。平 12 年 9 月より, 早稲田大学国際情報通信研究センター教授。ISO/IEC JTC 1/SC 29 議長。昭 63 本会学術奨励賞, 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。



岡田 重樹 (正員)

平 9 慶應大・理工・計測卒。平 11 同大学院計算機科学修士課程了。同年 NTT に入社。高効率通信のための画像符号化技術の研究開発に従事。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員。



小林 直樹 (正員)

昭 54 東工大・理・応用物理卒。昭 56 同大学院物理情報工学修士課程了。同年日本電信電話公社(現 NTT)に入社。以来, マルチメディア情報の入力技術, ヒューマンインタフェース設計技術, 画像符号化の研究開発に従事。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主幹研究員。