

情報源符号化部 H.264 | MPEG-4 AVC 規格の概要

平成 18 年 2 月 24 日

社団法人 電波産業会
デジタル放送システム開発部会
CS デジタル放送高度化作業班
映像符号化方式作業班

目 次

1. 標準化経緯	1
2. 方式の概要	2
3. 整数変換と量子化	5
3.1 整数変換	5
3.2 量子化	6
4. インター予測	7
4.1 複数参照フレーム	7
4.2 動き補償のブロック・サイズ	8
4.3 画素精度	8
4.4 動きベクトルの予測	9
4.5 重み付け予測	10
4.6 ダイレクトモード	11
5. イントラ予測	11
5.1 輝度信号のイントラ符号化	11
5.2 色差信号のイントラ予測符号化	13
6. エントロピー符号化	14
7. デブロッキング・フィルタ	15
7.1 デブロッキング・フィルタの役割	15
7.2 ブロック境界強度	16
7.3 フィルタリング処理	17
8. プロファイルとレベル	18
8.1 プロファイル	19
8.2 レベル	20
参考文献	21

1. 標準化経緯

H.264 | MPEG-4 AVC は MPEG-2 Video や MPEG-4 Visual に対してさらに効率の高い符号化技術を求める声に応じて新たに規格化された映像圧縮符号化方式である。標準化作業は ITU-T (International Telecommunications Union – Telecommunication Standardization Sector : 国際電気通信連合 電気通信標準化部門) の VCEG (Video Coding Experts Group : 映像符号化専門家グループ) と ISO/IEC (International Organization for Standardization : 国際標準化機構 / International Electrotechnical Commission : 国際電気標準会議) の MPEG (Moving Picture Experts Group : 動画像符号化専門家グループ) によって 2001 年 12 月に設置された JVT (Joint Video Team : 合同映像チーム) において進められた。

2 つの呼称は各機関におけるこれまでの映像圧縮符号化技術の勧告もしくは標準化作業を受け継いだものである。H.264 は 1990 年の H.261 に始まる ITU-T の映像圧縮符号化技術に係る勧告を指し、一方、MPEG-4 AVC は ISO と IEC が合同で設立した委員会 JTC1 (Joint Technical Committee 1 : 第 1 合同技術委員会) の作業グループの通称でもある MPEG の名で標準化された映像圧縮符号化方式を指す。さらに MPEG では MPEG-4 Visual の新たな符号化方式 “AVC : Advanced Video Coding” として ISO/IEC 14496-10 の規格番号が与えられたことから、MPEG-4 Part10 と呼ばれることもある。

MPEG では当初 MPEG-4 を MPEG-2 に対してより低いビットレート環境での符号化方式と位置付け、放送のみならず、WWW (World Wide Web) や双方向サービスにも高い適合性を持つ方式として標準化が進められた。MPEG-4 は 1998 年に初版、1999 年に第 2 版が作成されたが、時を同じくして、最新の符号化技術に MPEG-4 を超える性能を持つものが少なくないとの意見が強まり、新方式 (と同時に MPEG-4 を超える符号化効率を有する根拠の明示) の募集に至る。これが 1999 年 12 月の MPEG 会合における “Call for Evidence” である。

その後、応募方式に対する検討の結果、2000 年 7 月の MPEG 会合においてさらに具体評価に進むことが決定され、最終的には 2001 年 7 月に、新方式の標準化に十分な評価結果が得られたとして H.264 | MPEG-4 AVC の標準化作業が開始された。これと同時に方式の応募者でもあった ITU-T との合同作業が決定し、前述の JVT 設置を経て標準化作業が進められ、2003 年に初版が発行された。

また、2003 年の初版発行と同時に、より一層の高解像度化、高画質化を目的とした追加検討が開始された。初版が一般視聴を念頭に 4:2:0 映像信号の 8 ビットサンプリングを前提として標準化されたのに対し、ここでは 4:2:2 や 4:4:4 の映像信号を 10 ビットあるいは 12 ビットでサンプリングすることを狙いとし、業務用途の制作、伝送や HDTV 用の DVD、さらにはデジタルシネマ等への利用が目的とされた。この追加仕様については 2003 年 3 月の MPEG 会合で提案募集があり、初版で採用を見送った技術の見直しも併せてなされた結果、4:2:0 映像信号の 8 ビットサンプリングへのツール追加も含む形で 2004 年 7 月に標準

化が完了している。この追加仕様は Fidelity Range Extensions (FRExt : 高忠実度化規格) と呼ばれており、従来の Baseline、Main、Extended の 3 つのプロファイルに加え、新たに High、High 10、High 4:2:2、High 4:4:4 の 4 つのプロファイルが規定された。

2. 方式の概要

図 2-1 に H.264 | MPEG-4 AVC エンコーダの概略構成を示す。映像信号入力から、H.264 | MPEG-4 AVC ビットストリーム出力までが示されている。基本的なブロック構成は MPEG-2 に類似しており、符号化効率を上げるため、あるいは画質向上を目指して、一部機能が追加されるとともに、種々の工夫がなされている。図 2-1 に描かれた吹き出し項目は、主に MPEG-2 符号化方式などの従来方式にはない新たな要素技術である。

この章では、H.264 | MPEG-4 AVC 符号化を構成する主要な要素技術について、既存のデジタル放送で採用されている MPEG-2 などの従来の符号化方式と比較しながら簡単に説明し、さらに詳細については 3 章以降に記述する。

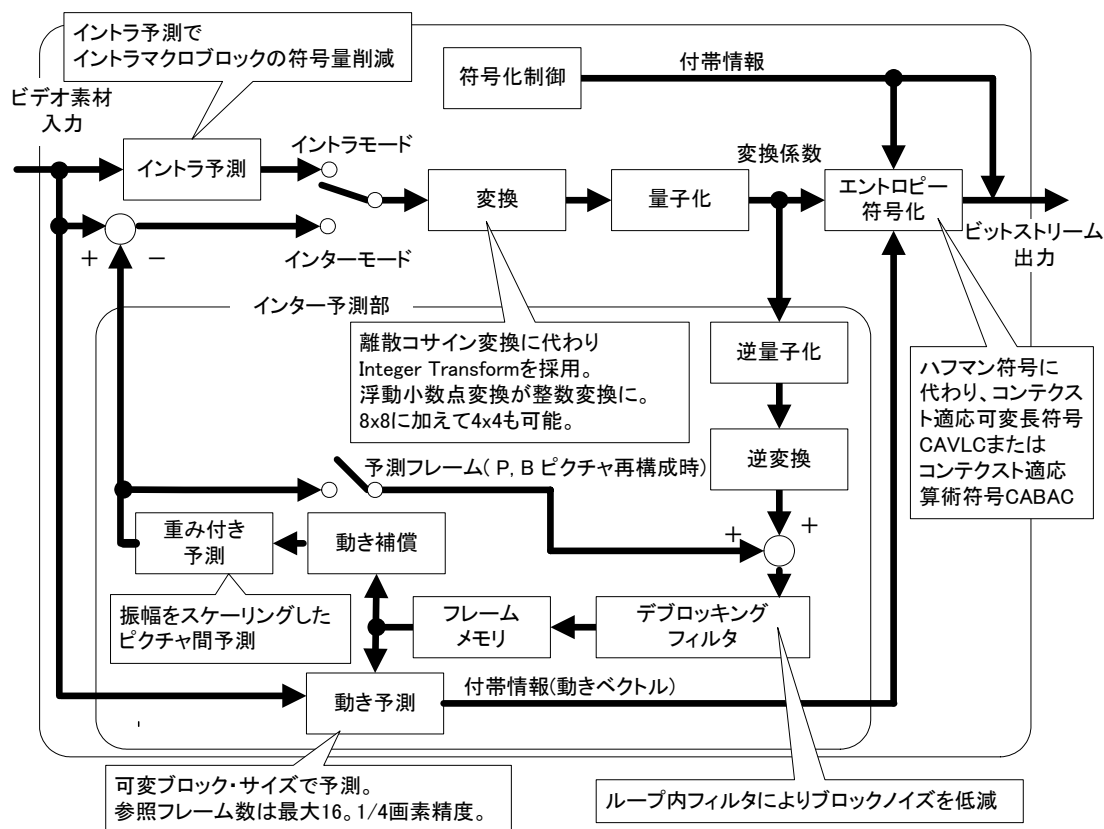


図 2-1 H.264 | MPEG-4 AVC エンコーダの概略構成

-変換

MPEG-2 では、8×8 離散コサイン変換が用いられていたのに対して、H.264 |

MPEG-4 AVC では 4×4 と 8×8 の 2 通りのブロック・サイズの Integer Transform (整数変換)を用いる。Integer Transform は離散コサイン変換の浮動小数点演算を整数化した手法である。ブロック・サイズとして 4×4 も選択可能になったことで、復号画像のブロック・ノイズを目立たなくすることができる。なお、 8×8 の Integer Transform は、High プロファイルなど FRExt で追加されたプロファイルでのみ使用可能である。

-インター予測

予測に用いるブロック・サイズは、従来の 16×16 や 8×8 だけではなく、最小 4×4 までの 7 種類のブロック形状で予測する。 16×16 ブロック内に複雑な部分と平坦な部分がある場合には、柔軟にブロック・サイズを変えて予測に用いることができるため、ピクチャ間で予測した差分値を削減できる。

また、H.264 | MPEG-4 AVC では、レベルによって異なるが最大 16 フレームを参照画像として用いることができる (HDTV に対応するレベル 4 の場合、最大 4 フレームを使用可能)。複数のフレームを用いることは演算量とメモリ量の増大につながるが、より類似したブロックを参照することが可能となり、差分データ値が小さくなり生成符号量を削減できる。

さらに、動き予測の精度が $1/4$ 画素精度に向上しており、MPEG-2 の $1/2$ 画素精度と比べて微小な動きを忠実に符号化できる。

予測の際は、振幅をスケールリングする重み付き予測を行うことで、時間的に明るさが変化する場合でも差分値を削減できる。

-イントラ予測

H.264 | MPEG-4 AVC から導入された方式で、符号化対象のイントラ・マクロブロックについてピクチャ内の近傍の復号済みの画素値から予測し、差分値を Integer Transform する。輝度信号の場合、 4×4 ブロックで 9 種類、 8×8 ブロックで 9 種類、 16×16 ブロックで 4 種類の予測モードを持つ。なお、 8×8 の Integer Transform は、High プロファイルなど FRExt で追加されたプロファイルでのみ使用可能である。

ブロックの画素値をそのまま離散コサイン変換(Discrete Cosine Transform)していた MPEG-2 の手法に比べて、予測により差分値自体が小さくなることと、多くの場合差分値データには原画像に比べて高周波成分が少なくなることから、量子化後の生成符号量を削減できる。

-エントロピー符号化

Baseline プロファイルおよび Extended プロファイルでは VLC(Variable Length Coding : 可変長符号化)ベースの符号を用いる。具体的には指数ゴロム符号(Exponential Golomb Coding)と、それをコンテキストに応じて切り替える CAVLC(Context-based Adaptive Variable Length Coding)である。指数ゴロム符号は MPEG-2 で使用されていたハフマン符号より処理が簡単である(ハフマン符号も可変長符号の一種である)。

Main プロファイルおよび High プロファイルなど FRExt で追加されたプロファイルでは、VLC ベースの CAVLC の代わりに、より圧縮効率の高い CABAC (Context-based Adaptive Binary Arithmetic Coding) という算術符号を用いることも可能であり、生成符号量をさらに削減できる。

-デブロッキング・フィルタ

エンコーダおよびデコーダ内において再構成画像作成時にデブロッキング・フィルタ (ループ内フィルタ) が用いられている。従来の MPEG-2 の手法にはなかったツール(H.261 などでは採用されていた)であり、演算量は増加するが復号画像のブロック・ノイズを低減することができる。

表 2-1 に、MPEG-2 の Main プロファイルと H.264 | MPEG-4 AVC の High プロファイルの要素技術比較表を示す。

表 2-1 MPEG-2 と H.264 | MPEG-4 AVC の要素技術比較表

要素技術	MPEG-2 (Main プロファイル)	H.264 MPEG-4 AVC (High プロファイル)
イントラ予測	なし	4×4 ブロックで 9 種類、 8×8 ブロックで 9 種類 ^(注1) 、 16×16 ブロックで 4 種類の 予測モード
インター予測	16×16 または 16×8 ブロック 直前および直後のフレームよ り予測 1/2 画素精度予測	16×16 を 7 種類のモードで分割 最大 16 フレーム ^(注2) より予測 1/4 画素精度予測
変換	8×8 離散コサイン変換	4×4 または 8×8 ^(注1) の Integer Transform
エントロピー符号化	ハフマン符号	コンテキスト適応可変長符号 (CAVLC)または コンテキスト適応算術符号 (CABAC) ^(注3)
デブロッキング・フィルタ	なし	あり

(注1) 8x8 は Main プロファイルでは使用不可

(注2) レベルによって異なり、レベル4の場合は最大 4 フレーム

(注3) CABAC は Main プロファイルおよび FRExt で使用可能

この章のまとめとして最後に、MPEG-2 と比較して、H.264 | MPEG-4 AVC のメリットとデメリットについて記述する。

-H.264 | MPEG-4 AVC のメリット

- 新規方式による生成符号量の削減
インター/イントラ予測や CABAC などの新規方式により生成符号量が削減できる。
- ブロック・ノイズの少ない復号画像
4×4 の Integer Transform によりブロック・サイズ自体が小さくな

ったことに加えて、再構成画像作成時にデブロッキング・フィルタが用いられることによりブロック・ノイズの少ない復号画像が得られる。

-H.264 | MPEG-4 AVC のデメリット

- 演算量の増大
高精度のインター/イントラ予測およびデブロッキング・フィルタなどの新規方式により演算量は増大する。
- メモリ量の増大
インター予測での参照フレーム数が増えたことなどにより、エンコーダやデコーダに必要なメモリ量が増大する。

3. 整数変換と量子化

3.1 整数変換

MPEG-2 では、 8×8 の実数精度 DCT (Discrete Cosine Transform : 離散コサイン変換) によって映像信号を周波数領域に変換 (直交変換) しているが、H.264 | MPEG-4 AVC では、 4×4 および 8×8 ^(注4) の整数精度の直交変換が用いられる。

直交変換が整数精度であるため、浮動小数点演算に似た演算のまるめ処理がなく、順変換と逆変換の不一致が生じず、かつ高速に処理できる。

H.264 | MPEG-4 AVC では、DHT (Discrete Hadamard Transform : 離散アダマール変換) と整数精度 DCT ^(注5) が使用され、イントラ (Intra : 画面内) / インター (Inter : 画面間) の予測方法や輝度/色差の信号内容により使い分けられる。

(1) マクロブロック (MB) が 16×16 ブロック単位のイントラ予測モードで符号化される場合は、輝度信号については、 4×4 の整数精度 DCT を行い、得られた 16 個の直流成分にさらに 4×4 の DHT を行う。したがって、この場合には 17 個のブロックを符号化することになる。

(2) 16×16 ブロック単位のイントラ予測モード以外で符号化される場合は、輝度信号については、MB を 16 個の 4×4 ブロックに分割し、各々のブロックを 4×4 の整数精度 DCT で直交変換する。

(3) 色差信号 Cb、Cr については、4:2:0 映像信号の場合、ブロック・サイズが縦横 1/2 (8×8) であるため、4 つの 4×4 ブロックに対して整数精度 DCT を行い、さらに直流成分に対して 2×2 の DHT を行う。FRExt においても、色差信号の直交変換は 4×4 単位である。

(注4) 8×8 の Integer Transform は、High プロファイルなど FRExt で追加されたプロファイルで適応的に使用可能である。

(注5) 正確には、整数演算を行うために DCT を近似したものであるが、本資料では「整数精度 DCT」と記述する。

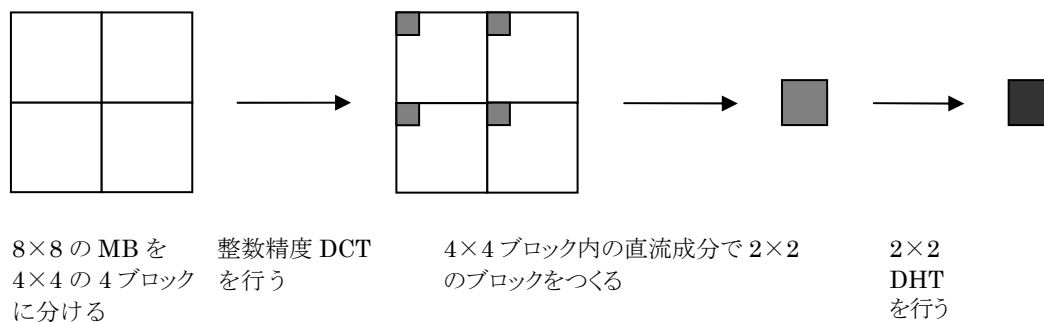
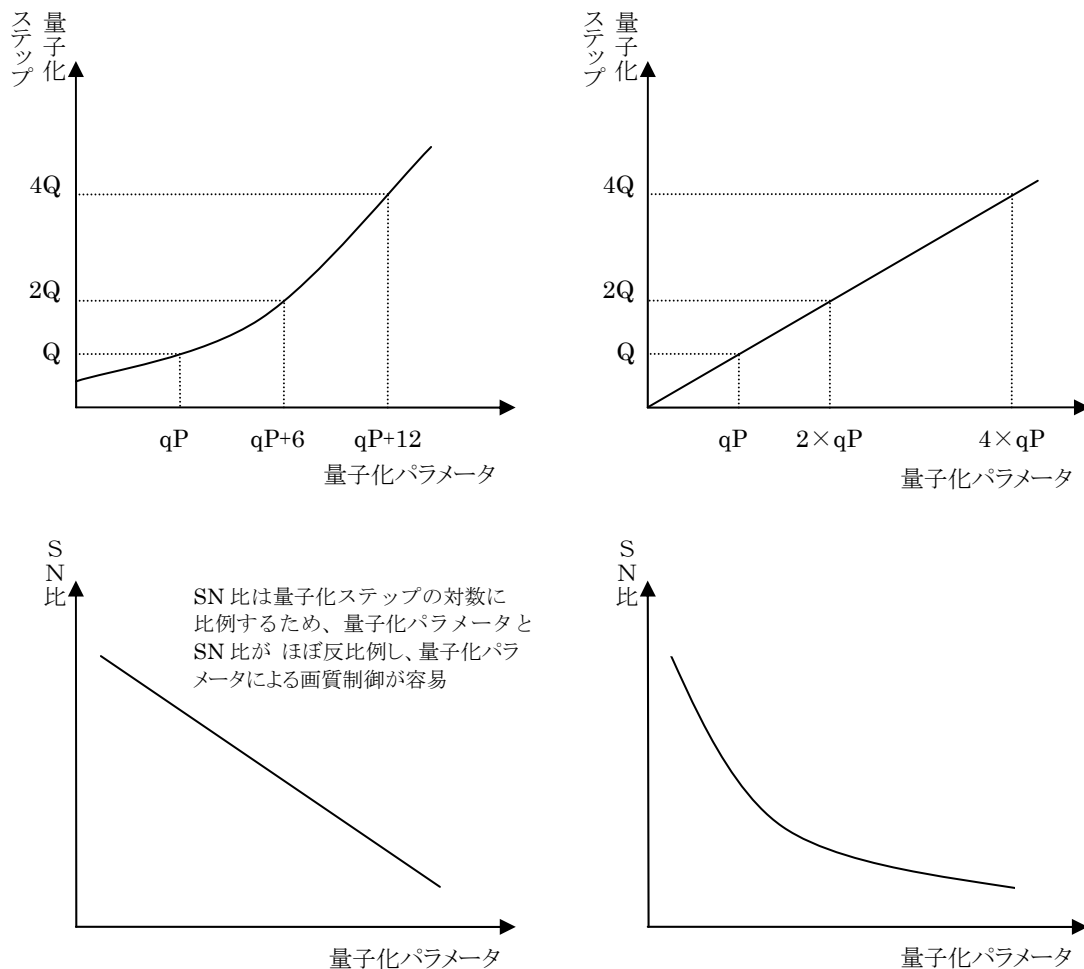


図 3-1 色差信号の変換

3.2 量子化

直交変換された信号を量子化して情報量を削減するが、H.264 | MPEG-4 AVC では、小さい成分を強制的に 0 に丸める領域（デッド・ゾーン）のない量子化処理をインター予測においても行っている。また、量子化パラメータが 6 増加する毎に量子化ステップが 2 倍となるようにして、量子化パラメータによる圧縮率の制御を容易としている。

なお、FRExt においては量子化スケーリング・リスト（量子化マトリクス）の使用が可能であり、画質劣化が目立ちにくい高周波成分を粗く量子化することが出来る。



H.264 | MPEG-4 AVC

MPEG-2

図 3-2 量子化特性

4. インター予測

H.264 | MPEG-4 AVC におけるインター予測（フレーム間予測）では、従来の MPEG-2 に比べて処理単位（ブロック・サイズ）を 4×4 まで小さくし、より細かな画素精度（ $1/4$ 画素まで）での動き補償をおこなっている。また、複数の参照フレームから最適なものを選択し、動き補償の効率を向上させている。さらに、MPEG-2 では画質劣化がみられたフェードインやフェードアウトなどの時間的に明るさの変化する画像に対する画質改善の手法も取り込まれている。

4.1 複数参照フレーム

MPEG-2 では、参照フレームに用いることができる画像は、P ピクチャの場合は直前の I ピクチャもしくは P ピクチャ 1 枚、B ピクチャの場合は過去、未来それぞれ 1 枚の I ピクチャもしくは P ピクチャで、過去と未来からの双方向

の予測が可能であった。H.264 | MPEG-4 AVC では、参照できるフレームの拡張を行い、過去、未来に関わらず 2 枚のフレームから予測（双予測）を行うことができるようになっている。予測画像を双予測ピクチャ（Bi-predictive Prediction-picture : B ピクチャ）と呼ぶ。

図 4-1 に従来の B ピクチャの参照方法を、図 4-2 に H.264 | MPEG-4 AVC での B ピクチャの参照方法を示す。

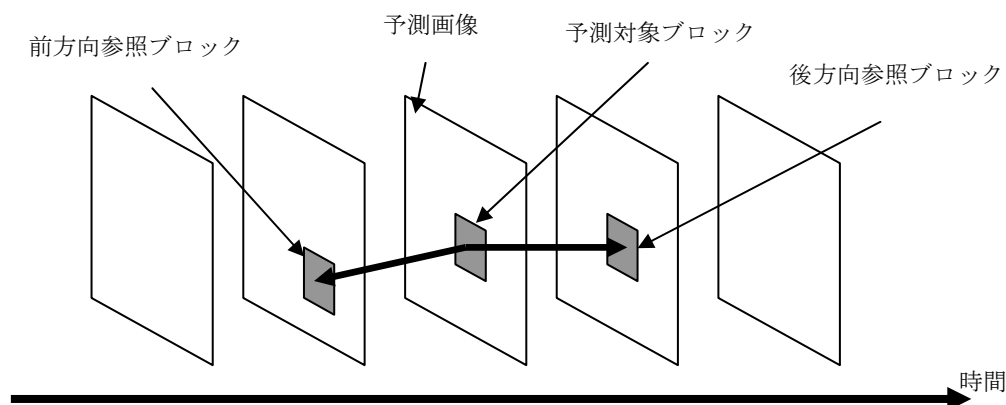


図 4-1 従来の B ピクチャの参照

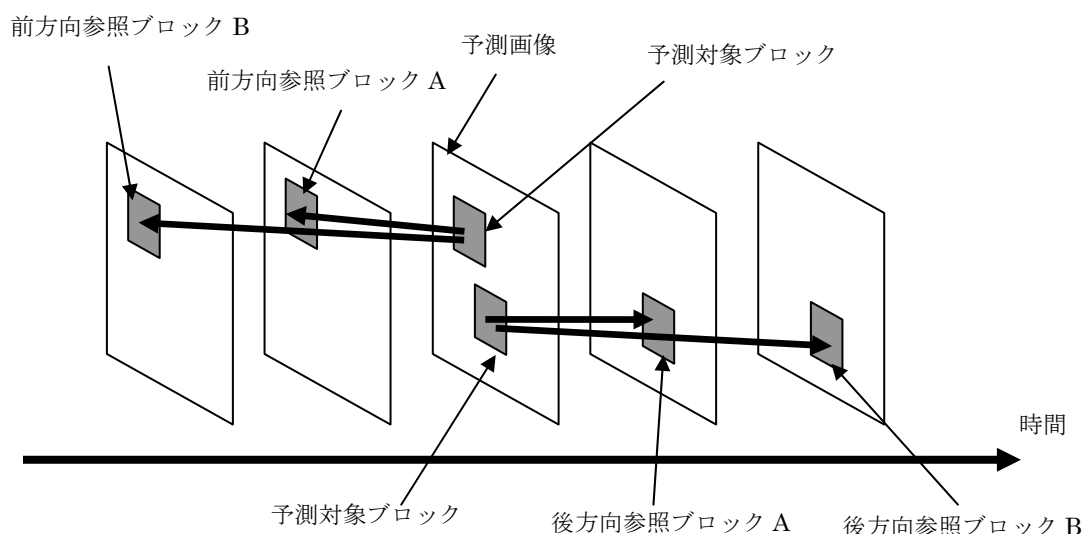


図 4-2 H.264 | MPEG-4 AVC の B ピクチャの参照

4.2 動き補償のブロック・サイズ

MPEG-2 では動き補償のマクロブロック・サイズは 16×16 画素に固定されていたが、H.264 | MPEG-4 AVC ではブロック・サイズを追加し、7 種類のブロック・サイズを用意し、任意に切り替えて用いることができる。動き補償に 16×16 以下のブロック・サイズを使用することで、従来より細かな動き補償を行うことができるようになっている。

4.3 画素精度

MPEG-2 では動き補償の画素精度は 1/2 画素であったが、H.264 | MPEG-4 AVC では参照ピクチャから 6 タップ FIR(Finite Impulse Response)フィルタを用いて 1/2 画素精度の予測画像を生成し、さらに 2 タップ平均値フィルタを用いて 1/4 画素精度の予測画像を生成するため、1/4 画素精度の動き補償が可能となっている。

表 4-1 動き補償のブロック・サイズと画素精度の比較

	MPEG-2	H.264 MPEG-4 AVC
ブロック・サイズ	16×16	16×16、16×8、8×16、8×8、8×4、4×8、4×4
画素精度	1/2	1/4

4.4 動きベクトルの予測

H.264 | MPEG-4 AVC における動きベクトルの予測は、従来の方法と同じく周辺ブロックからの中央値を用いて行っている。予測対象ブロックの左、上、右上の周辺ブロックから動きベクトルを予測する。

図 4-3 では予測対象ブロックの左ブロックを A、上を B、右上を C とし、それぞれの動きベクトルを MVa、MVb、MVc としたときの様子を表している。

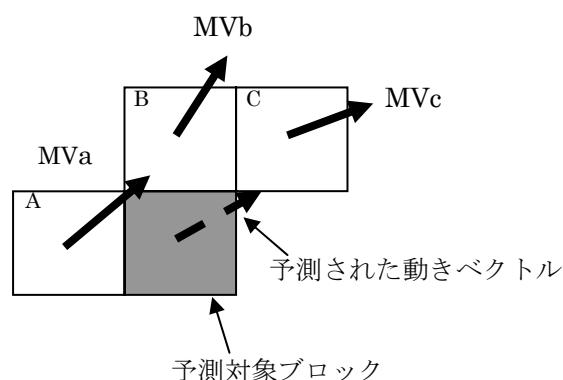


図 4-3 動きベクトルの予測

また、予測に用いるブロックは以下の規則に従って決定している。図 4-4 に例を示す。

- ・左に複数のブロックが隣接している場合は、その中の一番上のブロック(図 4-4 の A)を動きベクトルの予測に使用する。
- ・上に複数のブロックが隣接している場合は、その中の一番左のブロック(図 4-4 の B)を動きベクトルの予測に使用する。

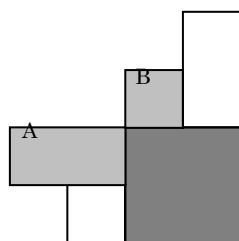


図 4-4 予測に用いるブロックの決定

なお、以下のような場合は特別な処理を行っている。

- 上のブロックと右上のブロックがピクチャやスライスの外にある時は、左のブロックの動きベクトルを用いる。
- 動きベクトルを予測する 3 つのブロックのうち、参照ピクチャが等しいブロックが 1 つだけある場合は、そのブロックの動きベクトルを用いる。
- 符号化するブロック・サイズが正方形でない場合は、周辺 3 つのブロックの動きベクトルの中央値ではなく、図 4-5 のように動きベクトルを予測する。
 - マクロブロック・サイズが 8×16 の場合(図 4-5 (a))、ブロック a はブロック A の動きベクトルを、ブロック b はブロック C の動きベクトルを予測に使用する。
 - マクロブロック・サイズが 16×8 の場合(図 4-5 (b))、ブロック a はブロック B の動きベクトルを、ブロック b はブロック A の動きベクトルを予測に使用する。

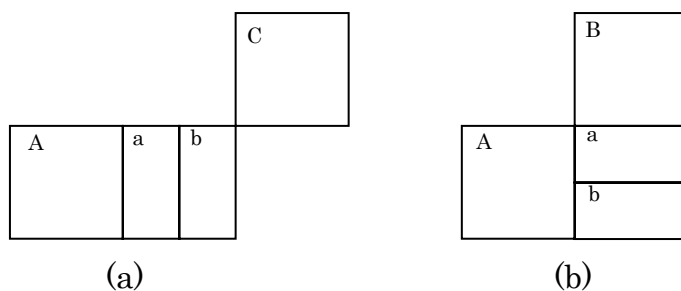


図 4-5 正方形でないブロックの動きベクトルの予測

4.5 重み付け予測

従来の方式では、画像の明るさを予測する手法がなかったため、フェードインやフェードアウトのように画像の明るさが時間によって変化するような画像では画質劣化があった。H.264 | MPEG-4 AVC では、参照ピクチャに重み係数を掛けて予測を行うことで画質の向上を行っている。

P スライスのインター予測と B スライスで参照フレームを 1 つだけ用いる予測では、動き補償予測信号に重み係数を掛けて予測信号の生成を行う。B スライスで参照フレームを 2 つ使う双予測では、2 つの動き補償予測信号にそれぞれ

重み係数を掛け、オフセット係数を足して予測信号の生成を行う。

また、B スライスでの重み付き予測では、Explicit(明示的)モードと Implicit(暗黙的)モードの 2 つの予測の種類があり、Explicit モードではスライスヘッダで重み係数とオフセット係数を送り、Implicit モードでは明示的に各係数を送るのではなく、参照フレームからの距離に応じて計算して係数を求める。

4.6 ダイレクトモード

H.264 | MPEG-4 AVC では、B ピクチャの圧縮効率を向上させるためにダイレクトモードを利用することができる。ダイレクトモードとは、動き情報を符号化済みのブロックの動き情報から予測を行う方法で、時間ダイレクトモード（異なる動きを含む画像で動きが一定の速度の場合に有効）と空間ダイレクトモード（同じような動きを含む画像で、動きが不規則な速度の場合に有効）の 2 方式がある。

5. イントラ予測

H.264 | MPEG-4 AVC では、フレーム間予測を用いずに符号化される画素ブロックに対して、符号化済みの隣接ブロックの画素値から予測画像を生成し、その予測画像との差分を符号化する画面内予測符号化（イントラ予測符号化）が採用されている。予測画像の生成単位となるブロック・サイズは、輝度信号については 4×4 画素のサブブロックおよび 16×16 画素のマクロブロックの 2 種類である。FRExt では、さらに 8×8 画素のブロック・サイズも導入され、3 種類の中から最適なブロック・サイズを選択できる。色差信号については常にマクロブロック・サイズの 1 種類である。また、予測画像生成における予測方向は、 4×4 および 8×8 輝度信号イントラ予測モードの場合はそれぞれ 9 種類、 16×16 輝度信号イントラ予測モードおよび色差信号イントラ予測モードの場合は 4 種類から選択できる。

MPEG-4 Visual で採用されているイントラ予測は、DCT 係数の直流成分と低周波の交流成分のみの予測により実現されていたのに対し、H.264 | MPEG-4 AVC では、DCT 係数ではなく画素レベルでの予測を行い、かつ垂直・水平方向以外にも斜め方向の予測モードも定義されており、さらに輝度信号 4×4 および 8×8 イントラ予測モードの場合にはサブブロックごとに最適な予測モードを選択して利用できるため、予測効率が大幅に向上している。

5.1 輝度信号のイントラ予測符号化

輝度信号の 4×4 画素単位のイントラ予測符号化では、図 5-1 に示すブロックを符号化する場合、符号化済みの左ブロックの中の 4 画素 (A、B、C、D)、上ブロックの中の 4 画素 (E、F、G、H)、右上ブロックの中の 4 画素 (I、J、K、L)、さらに左上ブロックの中の 1 画素 (M) の値から、符号化対象ブロック内の 4×4 画素の値を予測し符号化する。なお、参照する画素は、「デブロッキング・フィルタ (Deblocking filter)」を施す前の画素値を使用する。

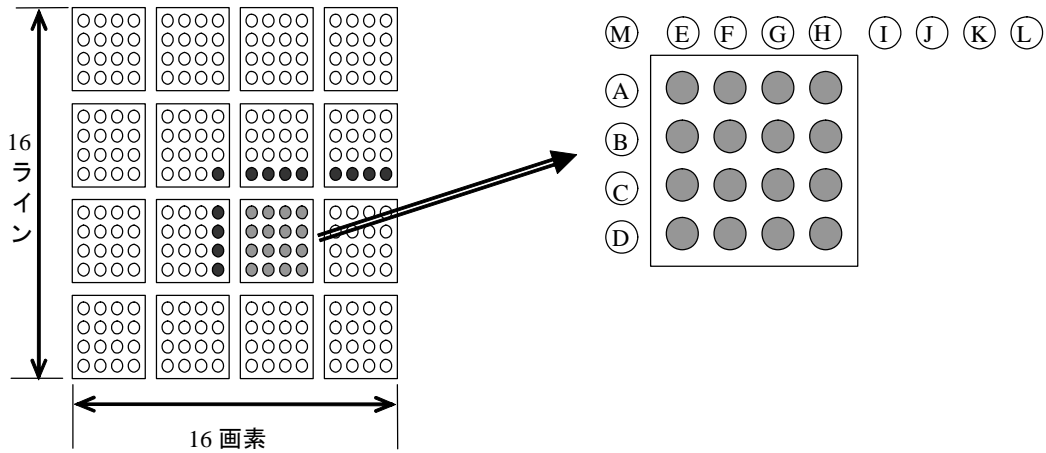


図 5-1 4×4 画素単位のイントラ予測符号化

予測方向は、平均値予測を含む図 5-2 の 9 通りの予測方向（予測モード 0～8）が定義され、発生頻度が高い予測方向順に小さい番号の予測モード番号が割り当てられている。例えば、予測モード 0 には上ブロックの画素を参照する垂直予測が、予測モード 1 には左ブロックの画素を参照する水平予測が定義されている。予測モード 2 の平均値予測は、左ブロックの 4 画素（図 5-1 の A、B、C、D）と、上ブロックの 4 画素（図 5-1 の E、F、G、H）の合計 8 画素の平均値で対象ブロックの画素全てを予測する。

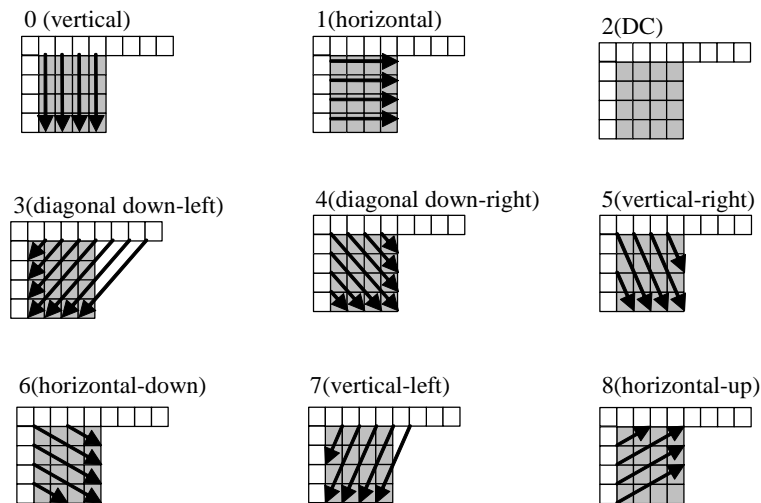


図 5-2 4×4 イントラ予測符号化の予測モード

マクロブロック内の 16 個のサブブロックそれぞれについて、この 9 通りの予測モードの中から最も適切に予測できる予測方向（予測モード）で符号化する。また、予測方向は隣接ブロックの予測方向と相関が高くなるという性質を利用して、符号化済みの隣接ブロックの予測方向を対象ブロックの予測方向の予測

値とする。その際、隣接ブロックから予測した予測方向と実際の対象ブロックの予測方向が同じ場合には、予測モード・フラグのみを符号化すればよく、予測方向の符号化に必要なビット数を節約できるので圧縮率が向上する。

また、FRExt では、高解像度映像の画質の再現性を高めるために 8×8 画素単位のイントラ予測も導入されている。予測方向は 4×4 の場合と同様の 9 通りである。

さらに、輝度信号の場合は 16×16 画素単位のイントラ予測符号化も利用できる。隣接する左、上、左上の符号化済みブロックの画素値で予測し、予測方向は発生頻度順に図 5-3 の垂直予測、水平予測、平均値予測、平面予測の 4 通りの予測モードが定義されている。予測モード 2 の平均値予測の場合、左ブロックの 16 画素と上ブロックの 16 画素の合計 32 画素の平均値で対象ブロックの画素全てを予測する。また、予測モード 3 の平面予測の場合、左、上、左上ブロックの画素を斜め方向に内挿処理を行ない予測する。

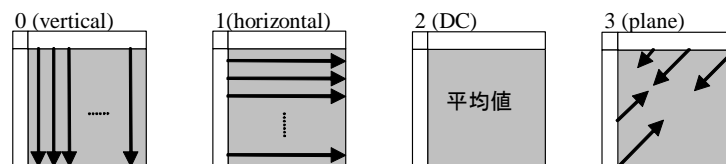


図 5-3 16×16 イントラ予測符号化の予測モード

16×16 イントラ予測の場合、予測方向はマクロブロック単位に 1 つであり、4×4 イントラ予測と比べて予測方向の符号化に必要なビット数が少なくて済むため、平坦な画像に対して使用すると大幅な圧縮が実現できる。

5.2 色差信号のイントラ予測符号化

色差信号のイントラ予測は常にマクロブロック・サイズ単位で行い、4:2:0 映像信号の場合には 8×8 画素単位で、FRExt で追加された 4:2:2 および 4:4:4 映像信号の場合には、それぞれ 8×16、16×16 画素単位で符号化する。予測方向は、16×16 輝度信号イントラ予測と同様に、符号化済みの左、上、左上の隣接ブロックの画素値を参照し、図 5-4 の 4 通りの予測モードが定義されている。但し、割り当てられている予測モード番号は 16×16 輝度信号イントラ予測とは異なり、発生頻度順に平均値予測、水平予測、垂直予測、平面予測の順に定義されている。なお、色差信号の予測方向は輝度信号とは独立に選択できるが、色差信号の Cb、Cr は同じ方向になる。

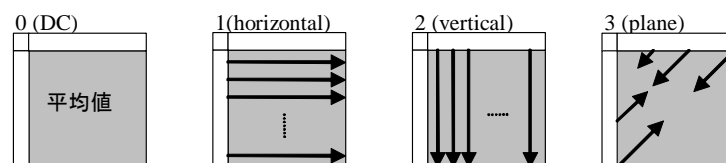


図 5-4 色差信号のイントラ予測符号化の予測モード

6. エントロピー符号化

H.264 | MPEG-4 AVC では、シンタックス要素（ヘッダ情報、動きベクトル、変換係数等）毎に異なるエントロピー符号化方式を用いる。

変換係数のように符号化データ中の大部分を占め、高い符号化効率及要求されるシンタックス要素においては、以下に示す 2 種類の符号化方式が規定されており、アプリケーションの特性に応じて、いずれか一方を選択して用いる。

- CAVLC (Context-based Adaptive Variable Length Coding : コンテキスト適応可変長符号化)
- CABAC (Context-based Adaptive Binary Arithmetic Coding : コンテキスト適応バイナリ算術符号化)

これらの方式は、周囲のブロックの状況（コンテキスト）に応じて、適応的に効率のよい符号語の割り当てを行うため、高い符号化効率を実現することができる。

CAVLC は変換係数にのみ使用され、周囲のブロックの状況に応じて、複数の可変長符号表の中から 1 つを選択し、この表を参照して各シンタックス要素に対応した符号語を割り当てる。図 6-1 に符号化対象ブロック C と、その周辺のブロックの位置関係を示す。ブロック A の非 0 の変換係数の個数を n_A 、ブロック B の非 0 の変換係数の個数を n_B とすると、その平均値 n_C の値に応じてブロック C の符号化に使用する可変長符号表を切り替える。

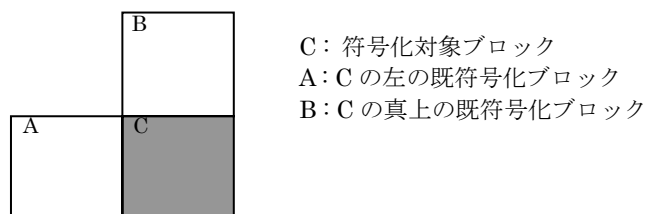


図 6-1 符号化対象ブロックとその周辺のブロック

表 6-1 に TotalCoeff と TrailingOnes というシンボルの組み合わせの符号化に使用する可変長符号表の一部を示す。表 6-1 では n_C の値に応じて 6 種類の可変長符号表が用意されている。例えば $n_A = 2$ 、 $n_B = 4$ の場合、 $n_C = 3$ となり、表 6-1 の網掛けした可変長符号表からブロック C の (TotalCoeff, TrailingOnes) に対応する可変長符号を選択する。

表 6-1 TotalCoeff と TrailingOnes の可変長符号表

TrailingOnes (coeff_token)	TotalCoeff (coeff_token)	$0 \leq nC < 2$	$2 \leq nC < 4$	$4 \leq nC < 8$	$8 \leq nC$	$nC == -1$	$nC == -2$
0	0	1	11	1111	0000 11	01	1
0	1	0001 01	0010 11	0011 11	0000 00	0001 11	0001 111
1	1	01	10	1110	0000 01	1	01
0	2	0000 0111	0001 11	0010 11	0001 00	0001 00	0001 110
1	2	0001 00	0011 1	0111 1	0001 01	0001 10	0001 101
2	2	001	011	1101	0001 10	001	001
0	3	0000 0011 1	0000 111	0010 00	0010 00	0000 11	0000 0011 1
1	3	0000 0110	0010 10	0110 0	0010 01	0000 011	0001 100
2	3	0000 101	0010 01	0111 0	0010 10	0000 010	0001 011
3	3	0001 1	0101	1100	0010 11	0001 01	0000 1
0	4	0000 0001 11	0000 0111	0001 111	0011 00	0000 10	0000 0011 0
.....							

CABAC は変換係数とヘッダ情報の一部に使用され、CAVLC が 1 個のシンタックス要素に 1 個の符号語を割り当てるのに対し、連続する複数個の同種のシンタックス要素に対し 1 個の符号を割り当てる効率の良い算術符号を用いる。また、シンタックス要素の出現頻度モデルを複数個準備しており、現在の符号化対象およびこの周囲の状況に応じて、適用するモデルを動的に切り替えながらシンタックス要素に符号を割り当てる。

CABAC を使用した場合、PSNR (Peak Signal to Noise Ratio : ピーク信号対雑音比) がおよそ 30~38dB の範囲 (比較的良好な画質) で CAVLC に比べ約 10%強の符号量の削減が可能である。その一方で、CABAC は膨大な演算量を必要とする。このような符号化効率と処理量のトレードオフの関係から、CABAC は符号化効率を優先する Main プロファイルや High プロファイルなど FRExt で追加された 4 種類のプロファイルでのみ使用できる。

一方、ヘッダ情報等のシンタックス要素では高い符号化効率が必須ではないため、比較的簡単な演算処理で符号化・復号処理が可能である指数ゴロム (Golomb) 符号を用いた符号化方式が採用されている。

7. デブロッキング・フィルタ

7.1 デブロッキング・フィルタの役割

ブロック単位の処理を用いる画像圧縮符号化方式には、ブロック境界に歪み (ブロック・ノイズ) が発生しやすいという欠点がある。特に MPEG-2 など従来の画像符号化方式においてインター予測を行う場合、動き補償時にブロック・ノイズを含んだ復号画像を参照してしまうことにより、フレーム間で画質

の劣化が伝播するという問題があった。

この問題を解決するために、H.264 | MPEG-4 AVC ではエンコーダおよびデコーダにデブロッキング・フィルタ (Deblocking Filter) を導入し、符号化によって生じたブロック・ノイズの補正を行っている。

デブロッキング・フィルタはループ内フィルタとして符号化ループに組み込まれており、復号画像に対して適応的に重み付けを行うことで予測信号を生成する。すなわち、ブロック全体を平滑化するフィルタとしてではなく、ブロック歪みの生じやすい箇所と生じにくい箇所とでフィルタの強度を変えるという適応制御を行っている。

このようなデブロッキング・フィルタを用いることにより、動き補償における予測誤差からブロック・ノイズの影響を除去し、符号化効率の向上を図ることができる。しかしながら、デブロッキング・フィルタには演算量が多いという欠点もある。このため、デブロッキング・フィルタ機能については ON/OFF を指定可能とし、ユーザが希望する場合にはデブロッキング・フィルタ機能を OFF にして処理量を削減するといった選択を可能としている。

7.2 ブロック境界強度

デブロッキング・フィルタは全てのブロック境界に均一にかけられるわけではなく、ブロック境界強度に応じてフィルタ適用画素およびフィルタの種類が決定される。図 7-1 に、フィルタリング処理前と処理後の画素値を示す。

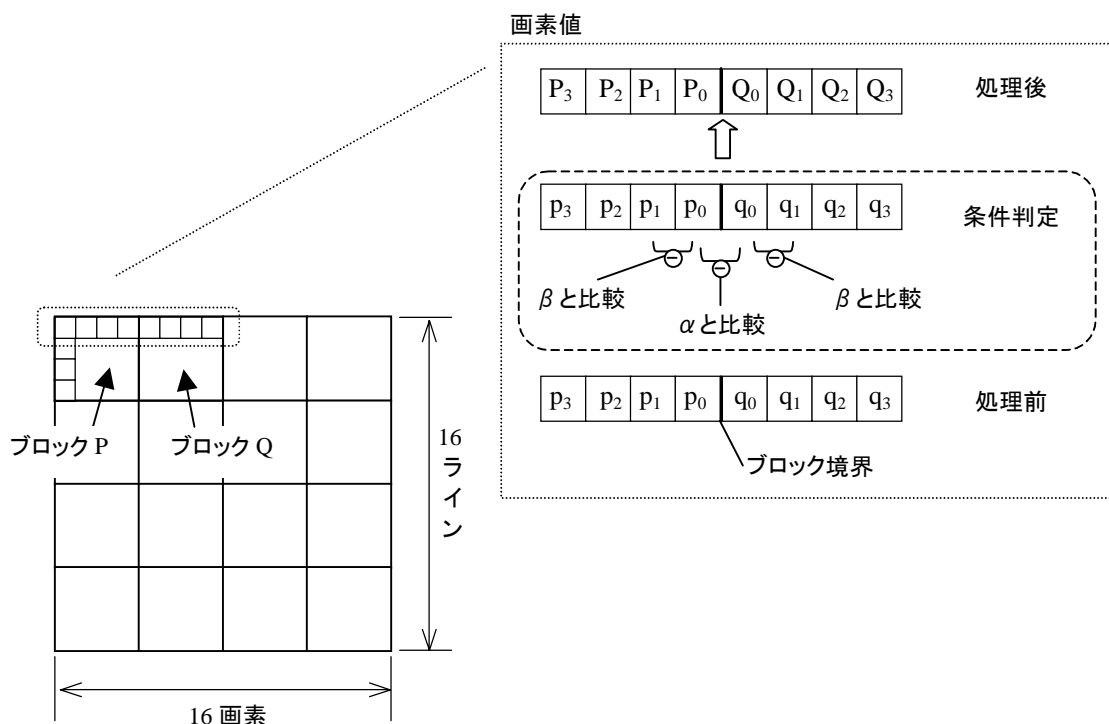


図 7-1 デブロッキング・フィルタ処理前と処理後の画素値

マクロブロックには、4×4 画素から成るブロック 16 個が含まれるが、ここ

ではブロック P とブロック Q について説明する。ブロック P とブロック Q における $p_0 \sim p_3$ および $q_0 \sim q_3$ はフィルタリング処理前の画素値である。この $p_0 \sim p_3$ と $q_0 \sim q_3$ により構成されるブロックに対して、画像のブロック境界の強度 (Bs 値 : Boundary Strength) が決定される。Bs 値の定義を表 7-1 に示す。

表 7-1 Bs 値 (Boundary Strength) の定義

Bs 値	p,q とイントラ・マクロブロックの関係	フィルタリング強度
4	<ul style="list-style-type: none"> ・p, q どちらかがイントラ・マクロブロックに属する ・マクロブロック境界に位置する 	最大
3	<ul style="list-style-type: none"> ・p, q どちらかがイントラ・マクロブロックに属する ・マクロブロック境界には位置しない 	
2	<ul style="list-style-type: none"> ・p, q どちらもイントラ・マクロブロックに属さない ・p, q どちらかのブロックが直交変換係数を持つ 	
1	<ul style="list-style-type: none"> ・p, q どちらもイントラ・マクロブロックに属さない ・p, q どちらのブロックも直交変換係数を持たない ・参照フレームか、動きベクトル値が異なる 	
0	<ul style="list-style-type: none"> ・p, q どちらもイントラ・マクロブロックに属さない ・p, q どちらのブロックも直交変換係数を持たない ・参照フレーム、動きベクトル値が同一 	なし

フィルタリング処理では、Bs 値が大きいほど強いフィルタリングを行う。すなわち、その境界がマクロブロック境界であるか否か、あるいはフレーム内予測を行っているか、などの条件に応じてフィルタリング強度を変えることにより、各々のブロック境界に適したフィルタリング処理を実行する。

7.3 フィルタリング処理

デブロッキング・フィルタは、表 7-1 に示す 5 通りの Bs 値に基づいてフィルタリング強度を調整している。具体的には、フィルタリング処理は次の条件が成立する場合にのみ実行 (ON) される。

(1) $Bs > 0$

(2) $|p_0 - q_0| < \alpha$; $|p_1 - p_0| < \beta$; $|q_1 - q_0| < \beta$

また、ブロック境界強度 (Bs 値) が 0 の場合、フィルタリング処理は行われない (OFF)。

図 7-1 に示すように、ブロック境界に位置する画素 p_0, q_0 についてはしきい値 α との比較を行い、ブロック内の p_0, p_1 および q_0, q_1 についてはしきい値 β との比較を行うことで、フィルタリングを適用するか否かが決定される。なお、しきい値 α, β は量子化パラメータ QP (Quantization parameter) に応じてその値が定められる。

また、しきい値 α, β については、ビットストリーム中のパラメータを操作することでオフセットを与えることが可能である。これにより、フィルタリング処理の ON/OFF に加え、ユーザはフィルタリング強度を段階的に調整することが可能となっている。

このように、画像の特徴に応じて適切なフィルタリング処理を行うことにより、ブロック・ノイズを抑制した良好な画素値 P0～P3 および Q0～Q3 を得ることができる。

8. プロファイルとレベル

H.264 | MPEG-4 AVC はモバイルからプロ・スタジオなどの広範な用途に適用できる高品質の汎用映像符号化方式であるが、デコーダに実装する機能や処理対象の映像等に制約を設けることにより、各用途において経済的、効率的な実現を可能とすると共に相互接続性を確保しており、この目的のために、プロファイルとレベルが規定されている。

プロファイルは、想定する主な用途に対応して 7 種類あり、各々において使用するツールと処理範囲等を定めている。レベルは、取り扱う映像の画素数に対応して 16 種類が定められている。

8.1 プロファイル

プロファイル	概要	主な仕様 ^(注6)	想定する主な用途
Baseline	双方向予測やインタレース映像への処理等を省いた基本的ツールで構成。 伝送エラー耐性用ツールを含む。	<ul style="list-style-type: none"> ・4:2:0 クロマフォーマット ・I, P スライスのみ ・重み付け予測^(注10)なし ・フレーム MB のみ ・算術符号化(CABAC)なし ・エラー耐性用ツール(FMO,ASO,RS^(注7)) 	モバイル放送、ネット配信など
Extended	Baseline プロファイルを含ると共に、より高画質・高度化のためのツールを追加。	<ul style="list-style-type: none"> ・4:2:0 クロマフォーマット ・I, P, B スライス ・重み付け予測^(注9)あり ・フレーム/フィールド適応 ・算術符号化(CABAC)なし ・Switching スライス(SI,SP スライス)^(注8) ・Slice Data Partition^(注9)あり 	高度な配信など
Main	標準的ツール群で構成される代表的なプロファイル。 FMO,ASO,RS ^(注7) を除く Baseline プロファイルを含む。	<ul style="list-style-type: none"> ・4:2:0 クロマフォーマット ・I, P, B スライス ・重み付け予測^(注10)あり ・フレーム/フィールド適応 ・算術符号化(CABAC)あり ・SI, SP スライスなし ・エラー耐性用ツール、Data Partition^(注9)なし 	放送、蓄積メディア、配信など
High	Main プロファイルを含ると共に、HDTV 映像等の高画質化用拡張(FRExt) ^(注12) ツール群を追加。	<ul style="list-style-type: none"> ・4:2:0 クロマフォーマット ・I, P, B スライス ・重み付け予測^(注10)あり ・フレーム/フィールド適応 ・8×8 画素直交変換あり ・周波数別重み付け量子化^(注11) ・算術符号化(CABAC)あり 	高画質放送、蓄積メディア、スタジオ(編集・蓄積)など
High 10	High プロファイルに 10 ビットサンプリング映像対応を追加。	<ul style="list-style-type: none"> ・4:2:0 クロマフォーマット ・10 ビットサンプリング映像対応 	素材配信、スタジオ(編集・蓄積)など
High 4:2:2	High 10 プロファイルに 4:2:2 クロマフォーマット対応を追加。	<ul style="list-style-type: none"> ・4:2:2 クロマフォーマット ・10 ビットサンプリング映像対応 	素材配信、スタジオ(編集・蓄積)など
High 4:4:4	High 4:2:2 プロファイルに 12 ビットサンプリング映像および 4:4:4 クロマフォーマット対応を追加。	<ul style="list-style-type: none"> ・4:4:4 クロマフォーマット ・12 ビットサンプリング映像対応 	素材配信、スタジオ(編集・蓄積)など

(注 6) 各プロファイルに共通する仕様として、イントラ予測、複数フレーム参照、デブロッキング(ループ内フィルタ)、可変ブロック・サイズ 1/4 画素精度動き補償、4×4 整数変換、CAVLC 等がある。

(注 7) FMO: Flexible Macroblock Ordering, ASO: Arbitrary Slice Order, RS: Redundant Slices

(注 8) SI,SP スライス: ビットストリーム間切り換えの高速化方法。

(注 9) Data partition: スライスデータを分割して伝送するエラー耐性強化方法。

(注 10) 重み付け予測: 複数参照画像を 1/2 以外の重み付けで加算し予測する Fade/Dissolve の強化方法。

(注 11) 周波数重み付け量子化: 直交変換係数の周波数別に重み付けを行う画質向上方法。

(注 12) FRExt : Fidelity Range Extensions

8.2 レベル

各レベルと主な制約

レベル	最大フレームサイズ (MBs)	最大MB処理レート (MB/s)	最大映像ビットレート (kbps)	動きベクトル		最大動ベクトル数 (連続する2個のMBあたり)	符号化ビクチャ	想定されるおもなプロファイル (下記範囲に限定されない)	
				水平成分範囲 (luma frame samples)	垂直成分範囲 (luma frame samples)				
1	99	1485	64	[-2048, +2047.75]	[-64, +63.75]	制限を置かない	フレームMBのみ	BP	
1b	99	1485	128		[-128, +127.75]				
1.1	396	3000	192						
1.2	396	6000	384						
1.3	396	11880	768						
2	396	11880	2000		[-256, +255.75]	32	フレームまたはフィールド	EP, MP	
2.1	792	19800	4000						
2.2	1620	20250	4000						
3	1620	40500	10000						
3.1	3600	108000	14000		[-512, +511.75]	16	フレームMBのみ	HP~	
3.2	5120	216000	20000						
4	8192	245760	20000						
4.1	8192	245760	50000						
4.2	8704	522240	50000						
5	22080	589824	135000						
5.1	36864	983040	240000						

各映像サイズに対応可能なレベルと最大フレームレート(fps)

映像フォーマット	Luma Width × Height	Max Frame Size (MBs)	レベル																
			1	1b	1.1	1.2	1.3	2	2.1	2.2	3	3.1	3.2	4	4.1	4.2	5	5.1	
SQCIF	128×96	48	30.9	30.9	62.5	125.0	172.0	172.0	172.0	172.0	172.0	172.0	172.0	172.0	172.0	172.0	172.0	172.0	
QCIF	176×144	99	15.0	15.0	30.3	60.6	120.0	120.0	172.0	172.0	172.0	172.0	172.0	172.0	172.0	172.0	172.0	172.0	
QVGA	320×240	300	/	/	/	/	10.0	20.0	39.6	39.6	66.0	67.5	135.0	172.0	172.0	172.0	172.0	172.0	
525 SIF	352×240	330	/	/	9.1	18.2	36.0	36.0	60.0	61.4	122.7	172.0	172.0	172.0	172.0	172.0	172.0	172.0	
CIF	352×288	396	/	/	7.6	15.2	30.0	30.0	50.0	51.1	102.3	172.0	172.0	172.0	172.0	172.0	172.0	172.0	
525 HHR	352×480	660	/	/	/	/	/	/	30.0	30.7	61.4	163.6	172.0	172.0	172.0	172.0	172.0	172.0	
625 HHR	352×576	792	/	/	/	/	/	/	25.0	25.6	51.1	136.4	172.0	172.0	172.0	172.0	172.0	172.0	
VGA	640×480	1200	/	/	/	/	/	/	/	16.9	33.8	90.0	172.0	172.0	172.0	172.0	172.0	172.0	
525 4SIF	704×480	1320	/	/	/	/	/	/	/	15.3	30.7	81.8	163.6	172.0	172.0	172.0	172.0	172.0	
525 SD	720×480	1350	/	/	/	/	/	/	/	15.0	30.0	80.0	160.0	172.0	172.0	172.0	172.0	172.0	
4CIF	704×576	1584	/	/	/	/	/	/	/	12.8	25.6	68.2	136.4	155.2	155.2	172.0	172.0	172.0	
625 SD	720×576	1620	/	/	/	/	/	/	/	12.5	25.0	66.7	133.3	151.7	151.7	172.0	172.0	172.0	
SVGA	800×600	1900	/	/	/	/	/	/	/	/	/	56.8	113.7	129.3	129.3	172.0	172.0	172.0	
XGA	1024×768	3072	/	/	/	/	/	/	/	35.2	70.3	80.0	80.0	80.0	170.0	172.0	172.0	172.0	
720p HD	1280×720	3600	/	/	/	/	/	/	/	/	/	30.0	60.0	68.3	68.3	145.1	163.8	172.0	
4VGA	1280×960	4800	/	/	/	/	/	/	/	/	/	/	45.0	51.2	51.2	108.8	122.9	172.0	
SXGA	1280×1024	5120	/	/	/	/	/	/	/	/	/	/	42.2	48.0	48.0	102.0	115.2	172.0	
525 16SIF	1408×960	5280	/	/	/	/	/	/	/	/	/	/	/	46.5	46.5	98.9	111.7	172.0	
16CIF	1408×1152	6336	/	/	/	/	/	/	/	/	/	/	/	/	38.8	38.8	82.4	93.1	155.2
4SVGA	1600×1200	7500	/	/	/	/	/	/	/	/	/	/	/	/	32.8	32.8	69.6	78.6	131.1
1080 HD	1920×1088	8160	/	/	/	/	/	/	/	/	/	/	/	30.1	30.1	64.0	72.3	120.5	
2Kx1K	2048×1024	8190	/	/	/	/	/	/	/	/	/	/	/	30.0	30.0	63.8	72.0	120.0	
2Kx1080	2048×1080	8704	/	/	/	/	/	/	/	/	/	/	/	/	/	/	60.0	67.8	112.9
4XGA	2048×1536	12288	/	/	/	/	/	/	/	/	/	/	/	/	/	/	/	48.0	80.0
16VGA	2560×1920	19200	/	/	/	/	/	/	/	/	/	/	/	/	/	/	/	30.7	51.2
3616x1536 (2.35:)	3616×1536	21696	/	/	/	/	/	/	/	/	/	/	/	/	/	/	/	27.2	45.3
3672x1536 (2.39:)	3680×1536	22080	/	/	/	/	/	/	/	/	/	/	/	/	/	/	/	26.7	44.5
4Kx2K	4096×2048	32768	/	/	/	/	/	/	/	/	/	/	/	/	/	/	/	/	30.0
4096x2304 (16:9)	4096×2304	36864	/	/	/	/	/	/	/	/	/	/	/	/	/	/	/	/	26.7

参考文献

- [1] ITU-T Recommendation H.264: Advanced video coding for generic audiovisual services (03/2005).
(H.264 (2005) Cor.1(09/2005)を含む)
- [2] 新井, 「H.264, あの手この手で SDTV/HDTV 録画を目指す」, 『日経エレクトロニクス』, 2005年1月3日号, pp.49-56.
- [3] 大久保榮監修 『H.264/AVC 教科書 改訂版』, インプレス、2006.
- [4] 亀山渉、花村剛監修 『デジタル放送教科書 (上) 改訂版』 インプレス、2004.
- [5] vcodex : H.264 tutorial white papers, <http://www.vcodex.com/h264.html>,
「H.264 Reconstruction Filter.pdf」
- [6] 放送技術開発協議会 TV 情報符号化委員会 高能率符号化作業班、『MPEG-2 規格の概要 (Video) : MPEG 符号化要素技術』 1994年9月26日.
- [7] Iain E G Richardson, 「H.264 and MPEG-4 Video Compression」, John Wiley & Sons, Autumn 2003.

MPEG-2規格の概要 (Video)

MPEG符号化要素技術

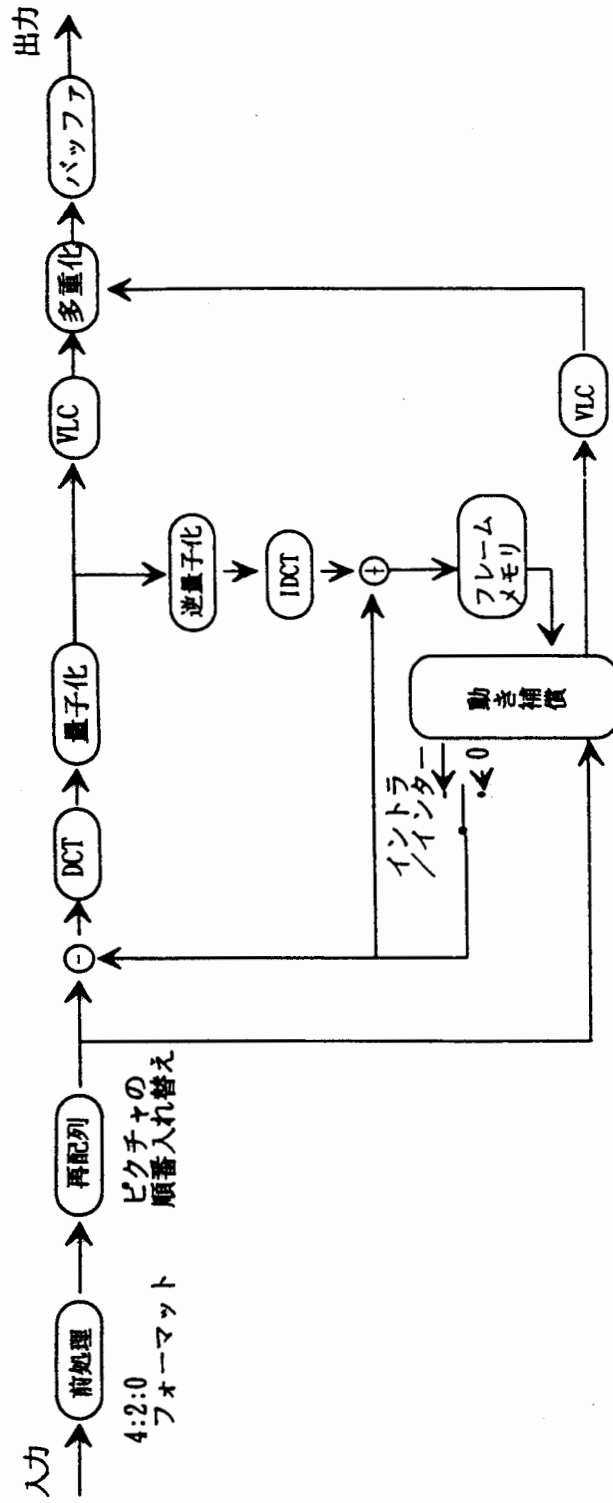
1994年9月26日

放送技術開発協議会
TV情報符号化委員会
高能率符号化方式作業班

目次

1. ビデオデータの処理単位	165
2. ピクチャ構造	167
3. DCT処理	171
4. 量子化	175
5. 動き検出	180
6. 動き補償	182
7. 可変長符号	185
8. レートコントロール	187
9. 前処理/後処理	190
9.1 4:2:0フォーマット	190
9.2 3/2プルダウン	193
9.3 パン/スキャン	194
10. スケーラビリティ (階層構造)	195
11. ProfileとLevel	198
12. シンタックス	201
13. エラー対策 (耐性)	205
14. 蓄積系、コンピュータ応用等	207
14.1 MPEGのビデオテープレコーダ記録	207
14.2 一般蓄積系・コンピュータ応用に対するMPEG2の考え方	209
15. その他	211
15.1 データパーティショニング	211
15.2 ACリーク	213
15.3 ローデイレイモード	214
15.4 タイムスタンプ	215
参考文献	216
付録	217
ビデオ・シーケンス層	217
GOP層	220
ピクチャ層	220
スライス層	222
マクロブロック層	223
ブロック層	224
略号一覧	225
索引	226

MPEG2 符号化 (Main Profile) の構成



1. ビデオデータの処理単位

【概要】

MPEG2のデータ単位は図1に示す6層の階層構造を持つ。

各データ単位の特徴及び処理は以下の通り。

1. ビデオシーケンス (Video Sequence)

最も上位のデータ単位。その動画像全体に渡って有効なパラメータ(画像のサイズ、ビットレート等)がヘッダとして伝送される。

2. GOP (Group of Picture)

最低1つのIピクチャを含む複数のピクチャから成るデータ単位。ランダムアクセスや動画編集の単位として用いられることを想定し、そのGOP内のBピクチャがGOP内の閉じた処理で復号可能か否かのフラグやタイム・コード等の情報をヘッダに含む。

3. ピクチャ (Picture)

1フレームあるいは1フィールドに当たるデータ単位。フレーム間予測方法の許容度に応じたI/P/Bピクチャの種別がある。Bピクチャを用いる場合、エンコーダ(及びデコーダ)でピクチャの入力(出力)順と符号化(復号化)順が異なるため、ピクチャの順番入れ換え(これらの操作をordering/reorderingと呼ぶ)が必要となる。

4. スライス (Slice)

複数(最低1つ)のマクロブロックから成るデータ単位。同じ水平マクロブロックラインで完結しなくてはならないという制限がある。DSM(Digital Storage Media)応用での高速再生への適用を目的としたイントラスライスフラグをヘッダに挿入できる。

イントラスライス … イントラマクロブロックのみで構成されたスライス

5. マクロブロック (Macroblock)

16×16画素の領域に当たる4個のY信号ブロックとそれに対応するCr/Cb信号のブロックから成るデータ単位(4:2:0の場合は計6ブロック)。動きベクトルを用いた動き補償やフレーム/フィールドDCTの切り替えは基本的にこの単位で行われる。情報を伝送する必要のないマクロブロックはビットストリームに出力されない。

6. ブロック (Block)

8×8画素のデータからなる最下層のデータ単位。DCTやジグザグスキャンはブロック単位で行われる。

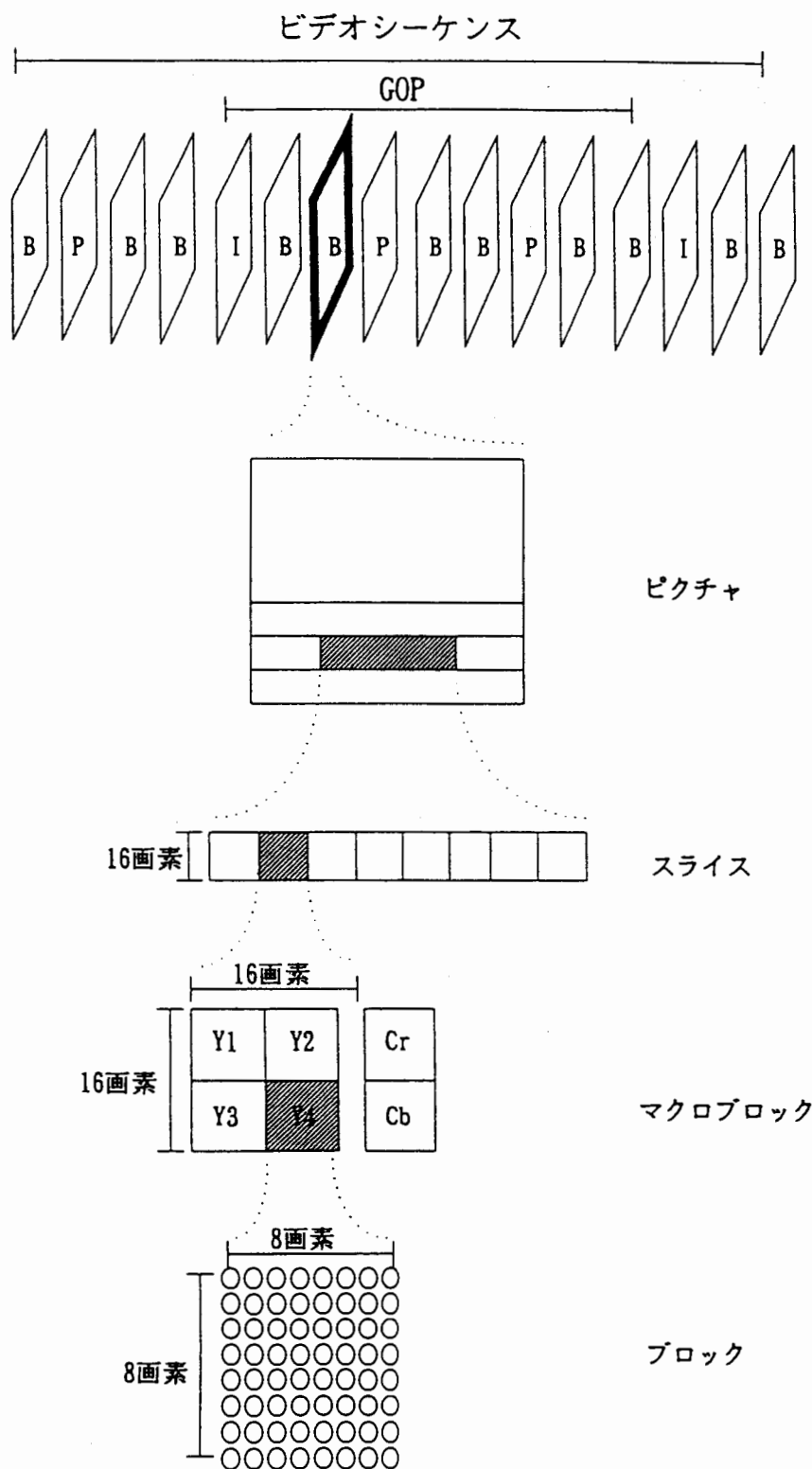


図1 MPEG2のデータ構造

【2次分配に関連する特徴】

- ・通信/放送系などで伝送途中からの受信/再生を可能とするため、シーケンスヘッダは（量子化マトリクス例外を除いて）同じ値のものを途中で複数回伝送することができる。
- ・ランダムアクセスや動画編集を重要視しないアプリケーションにおける効率改善のために、GOPレイヤはオプション（GOPヘッダは伝送しなくてもよい）となっている。

2. ピクチャ構造

【概要】

- ・ 3種類のピクチャと符号化モードがある。
 - Iピクチャ : intra modeのみ
 - Pピクチャ : intra modeとI/Pピクチャからのforward prediction
 - Bピクチャ : intra modeとI/Pピクチャからのforward prediction
/ backward prediction / bi-directional prediction
- ・ Iピクチャを少なくとも一枚含むピクチャの集合をGOPと称し、編集、ランダムアクセスの単位となる。
- ・ 典型的なGOP構造の例を図2-1に示す。
- ・ GOP構造の周期は以下のパラメータで記述される。
 - N : Iピクチャの周期 (ピクチャ数) = GOPの長さ
 - M : I/Pピクチャの周期 (ピクチャ数)
- ・ Bピクチャを含む場合、エンコーダ/デコーダでそれぞれピクチャの順番を入れ替える処理 (ordering/reordering) が必要になる。(図2-2)

【GOP構造の比較】

- ・ 図2-1の各typeのGOP構造を以下の観点から比較する。
- ・ レート対画質 (筆者の主観によるITU-R Rec. 601近似レベル到達ビットレート)
 - Type 1 (I) : 24 Mbps
 - Type 2 (I/P) : 12 Mbps
 - Type 3 (I/P/B) : 8 Mbps
 - Type 4 (P) : 14 Mbps
 - Type 5 (P/B) : 10 Mbps

ハードウェア規模

- Type 1 (I) : 小規模 (ME/MC回路なし)
frame memory 0 frame
- Type 2 (I/P) : 大規模
frame memory 1 frame
- Type 3 (I/P/B): 非常に大規模 (特にME回路が大きい)
frame memory 2 frame
- Type 4 (P) : Type 2 とほぼ同様
- Type 5 (P/B) : Type 3 とほぼ同様

遅延時間 (ビットレート:8Mbps での実現可能な最小遅延時間(フレーム数))

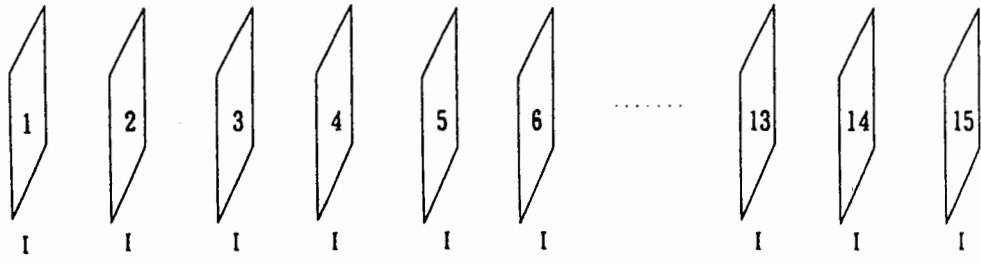
- Type 1 (I) : ほぼ 0 フレーム
- Type 2 (I/P) : 2 ~ 3 フレーム
- Type 3 (I/P/B): 10 フレーム
- Type 4 (P) : 2 ~ 3 フレーム
- Type 5 (P/B) : 6 ~ 7 フレーム

誤り伝般

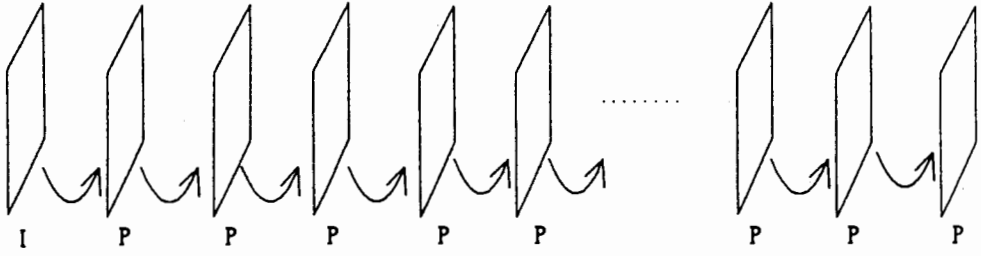
- Type 1 (I) : 1 ピクチャ
- Type 2 (I/P) : 最大 1 GOP (どの誤りも次のI ピクチャまで伝般)
- Type 3 (I/P/B): 最大 1 GOP (Bピクチャの誤りは伝般しない)
- Type 4 (P) : 最大 1 GOP
- Type 5 (P/B) : 最大 1 GOP

編集単位

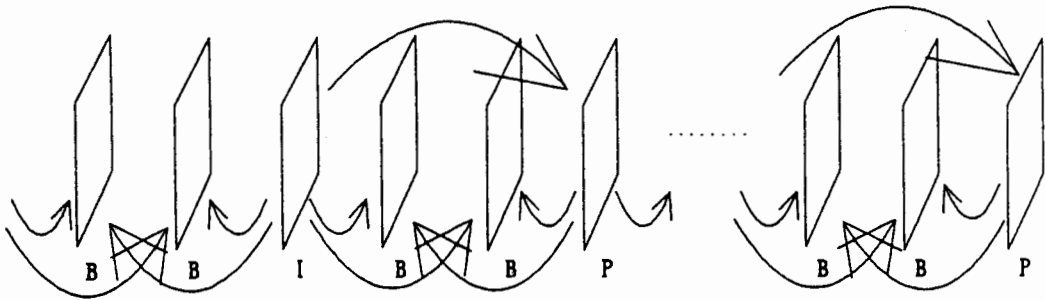
- Type 1 (I) : ピクチャ
- Type 2 (I/P) : GOP
- Type 3 (I/P/B): GOP (但しclosed GOPであること)
- Type 4 (P) : 編集不可能
- Type 5 (P/B) : 編集不可能



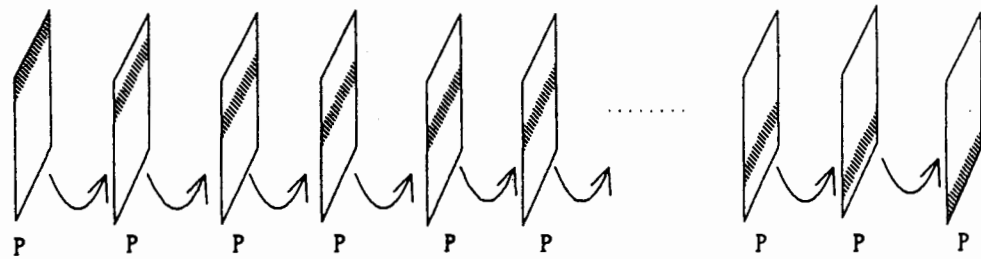
(a) Type1 (Iピクチャのみ)



(b) Type2 (IピクチャとPピクチャ)

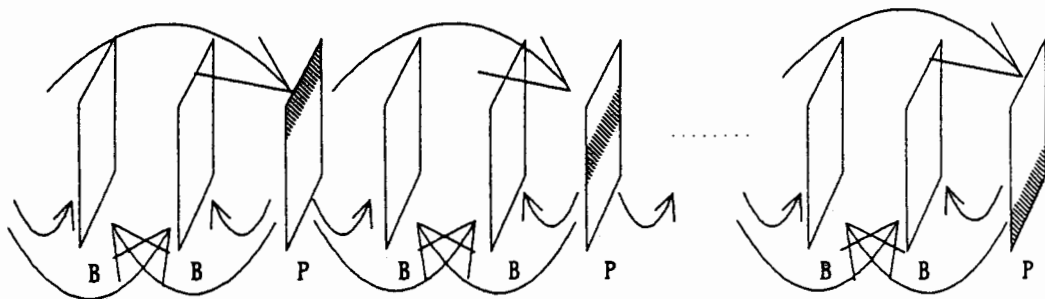


(c) Type3 (IピクチャとPピクチャとBピクチャ)



(d) Type4 (Pピクチャのみ)

/// はイントラスライス



(e) Type3 (PピクチャとBピクチャ)

図2-1 GOP構造の例

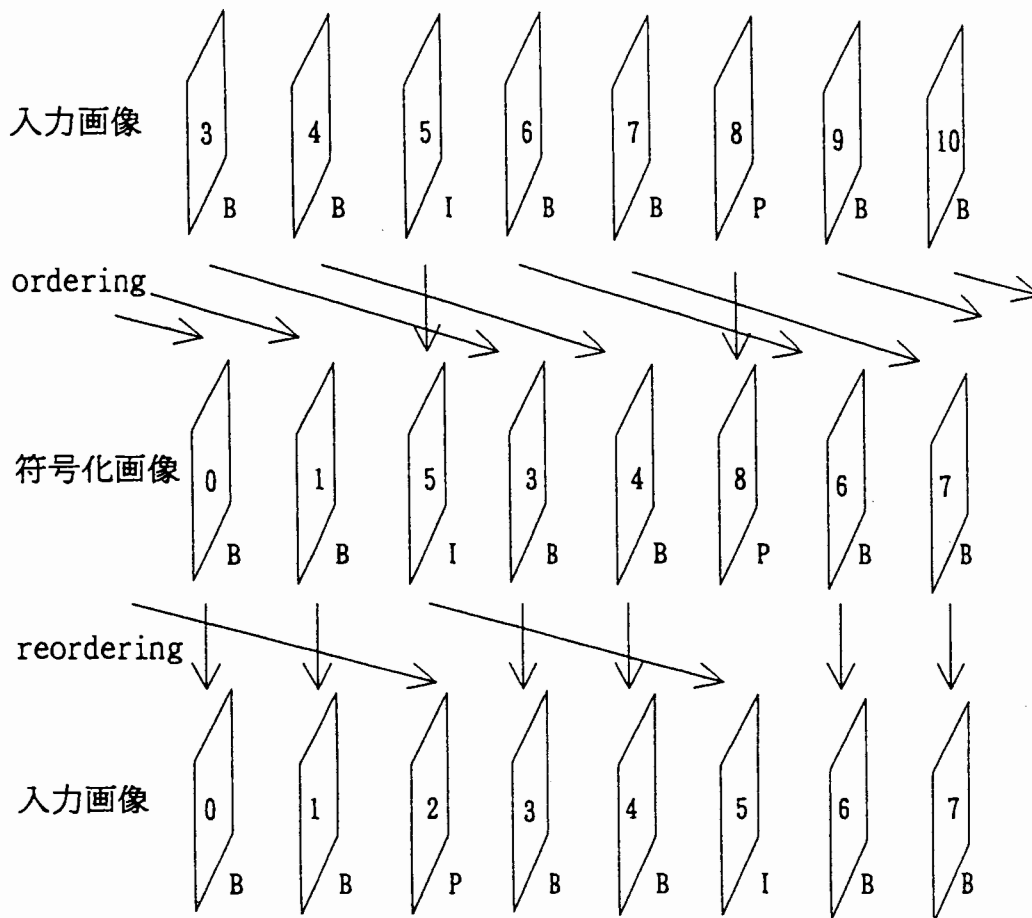


図2-2 フレーム順序の並べ換え

【クローズドGOP】

- ・Iピクチャで始まり、IピクチャまたはPピクチャで終わるGOPをクローズドGOPと呼ぶ。
- ・クローズドGOPの復号化には他のGOPが不要であり、GOP単位の編集でピクチャの消失を防ぐためにはクローズドGOPでなければならない。
- ・クローズドGOPでないGOP (Bピクチャで終わるGOP) では、編集などにより後続のGOPが消失すれば最後のBフレームが正しく復号化できないので、消失していることを示すフラグ（ブローケン・リンク）が附加される。

3. DCT処理

【DCTの概要】

- ・ 離散コサイン変換 (Discrete Cosine Transform) の略
- ・ $N \times N$ の画素 $f(j, k)$ に対する2次元DCT係数 $F(u, v)$ の変換式は以下のとおり

$$F(u, v) = \frac{4C(u)C(v)}{2N} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} f(j, k) \cos\left\{\frac{(2j+1)u\pi}{2N}\right\} \cos\left\{\frac{(2k+1)v\pi}{2N}\right\}$$

$$\text{ただし、} C(u) = \begin{cases} 1/\sqrt{2} & \text{for } u = 0 \\ 1 & \text{for } u \neq 0 \end{cases}$$

- ・ ITU-R Rec. 723/ITU-T H. 261/JPEG/MPEGで使用されている。
- ・ サイズ N を8に選んだ 8×8 DCTが一般的である。
- ・ $N \times N$ 画素のブロックの中だけで完結するブロック処理である。

DCTの長所

- ・ 自然画に対する変換効率 (特定の係数への電力集中特性) が非常に高い。
- ・ ブロック処理であるので、変換や量子化による影響がブロック外に伝搬しない。
- ・ また、動き補償もブロック単位で行うのでこれと相性が良い。
- ・ また、各種の適応処理の切り替えをブロック単位で行うことが出来る。
- ・ 高速アルゴリズムが存在する。
- ・ 既に多くのLSIがつくられている。

DCTの短所

- ・ 実数乗算が必要である。
- ・ 計算誤差の影響により、可逆変換は実現不可能。
- ・ 圧縮率をあげるとブロック歪み、モスキート雑音を発生する。

【DCTとDPCMの比較】

DCTの長所

- ・ 低ビットレートで符号化効率が高い。
- ・ 視覚特性 (空間周波数) に応じて量子化特性を設定できる。

DCTの短所

- ・ 可逆符号化が難しい。

【DCTとWaveletの比較】

DCTの長所

- ・ブロック単位で処理ができるので、ブロック単位の処理である動き補償や量子化等の適応処理と相性が良い。

DCTの短所

- ・ブロック歪み、高周波のモスキート雑音による画質劣化がある。
- ・高品質に多重解像度で復号することは困難。

【フレーム／フィールド適応DCTの概要】

- ・動きが激しい部分ではフィールドで、動きが少ない所ではフレームで適応的にブロック化を行うことにより、垂直方向の相関が高い状態にしたうえでDCT変換を行う。
- ・マクロブロック単位で適応化の切り替えを行う。
- ・適応化は輝度信号のみで行い、色差信号は8×8の固定のDCTを行う。
- ・フィールドDCTとは、図3-1に示すように片方のフィールドに属する画素のみから8×8画素のブロックを構成しDCT変換を行うものである。
- ・フレームDCTとは、図3-2に示すように両方のフィールドから空間的に隣接する8×8画素のブロックを構成しDCT変換を行うものである。
- ・垂直の相関性がフレーム／フィールドの違いにより変わるので、変換係数の分散や視覚的重みに変化する。従って、量子化テーブルやVLCも切り替えた方がよいと思われる。

長所

- ・インターレース映像において、動静どちらの領域でも変換効率を高く維持できる。
- ・2次元DCT変換部分はそのままで、動静に対応できる。

短所

- ・ブロックをフレーム／フィールドに再構成するためのハードウェアが増加する。
- ・エンコーダではフレーム／フィールドの判定機構が必要になる。
- ・マクロブロックあたり1ビットのオーバーヘッドを生ずる。
- ・フィールド／フレームのブロック機構により、フィールド処理のみの場合と比較し、送受で1フィールド遅延が増える。

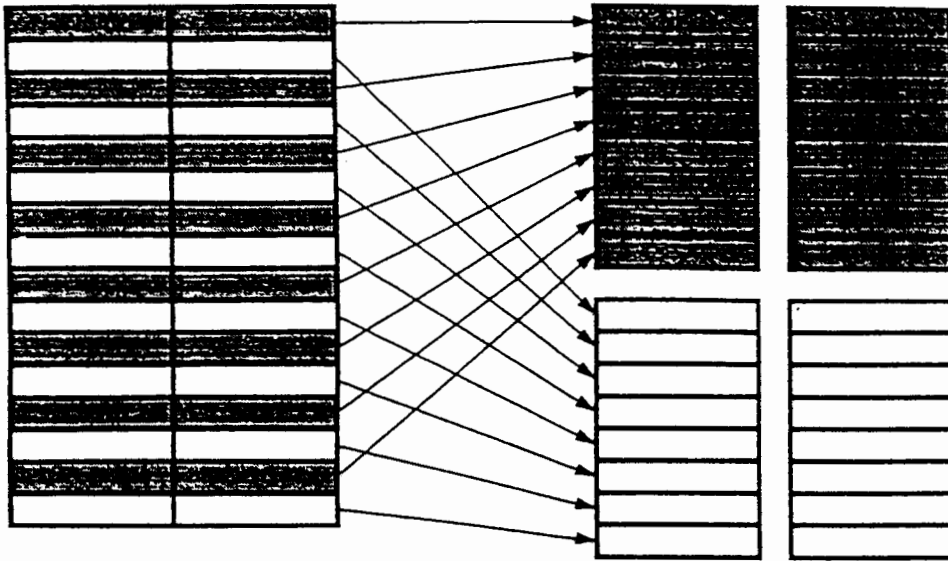


図3-1 フィールド内ブロック化

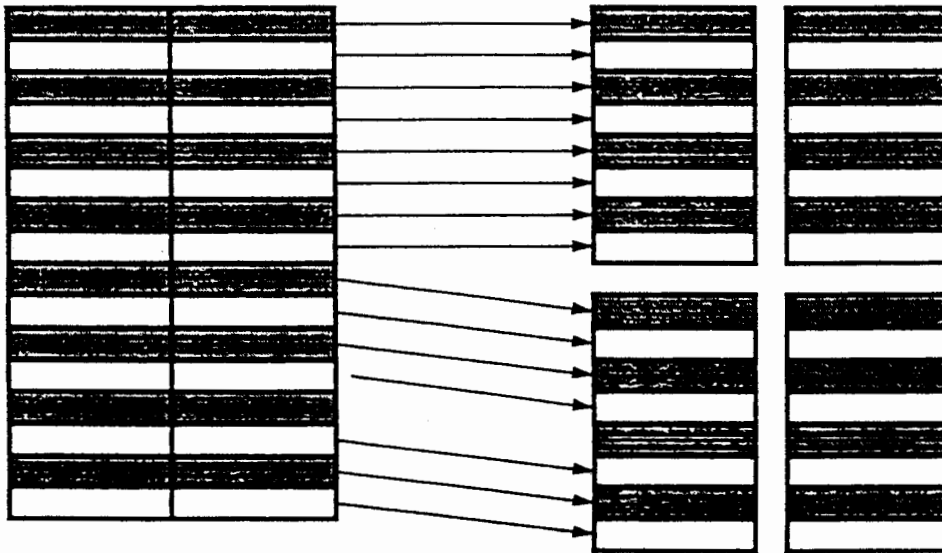


図3-2 フレーム内ブロック化

【適応ジグザグスキャン】

輝度信号は、フィールド内/フレーム内適応ブロック化されるが、フィールド内ブロック化の方がフレーム内ブロック化よりも垂直空間周波数が低いので、DCT係数のエネルギーも垂直周波数が低い成分に集中する。そこで、エネルギー分布に対応してジグザグスキャンの順番を変更すれば、符号化効率が向上する。MPEG2では、2通りのジグザグスキャンを準備し、ピクチャ毎に一方を選択できるようになっている。

図3-3と図3-4に2次元DCT変換成分のジグザグスキャン(符号化順序)を示す。

	0	1	2	3	4	5	6	7
0	0	1	5	6	14	15	27	28
1	2	4	7	13	16	26	29	42
2	3	8	12	17	25	30	41	43
3	9	11	18	24	31	40	44	53
4	10	19	23	32	39	45	52	54
5	20	22	33	38	46	51	55	60
6	21	34	37	47	50	56	59	61
7	35	36	48	49	57	58	62	63

図3-3 ジグザグスキャン(その1)

	0	1	2	3	4	5	6	7
0	0	4	6	20	22	36	38	52
1	1	5	7	21	23	37	39	53
2	2	8	19	24	34	40	50	54
3	3	9	18	25	35	41	51	55
4	10	17	26	30	42	46	56	60
5	11	16	27	31	43	47	57	61
6	12	15	28	32	44	48	58	62
7	13	14	29	33	45	49	59	63

図3-4 ジグザグスキャン(その2)

4. 量子化

【概要】

量子化はDCT変換係数の全体を小さい数で表現し、小さい数に短い符号を割り当てて符号量を減少させることを目的としている。2次元DCT変換係数の対応する位置に重みづけした量子化マトリクスで量子化される。また、量子化マトリクスには、画像用と差分画データ用の2種類あり各々に対応する、イントラ用とインター用デフォルト量子化マトリクスが用意されている。量子化の方法は、イントラはDCTのDC係数とAC係数で方法が異り、DC係数は量子化係数のみで量子化されるが、AC係数は量子化係数と量子化スケール幅係数により量子化される、そしてインターではDC項AC項とも量子化係数と量子化スケール幅係数で量子化が行われる。

【デフォルト量子化マトリクス】

(a) イントラ/インターでのデフォルトの意味

量子化を効率的に行うためにイントラ用（表4-1）とインター用（表4-2）の2種類のデフォルト量子化マトリクスが用意されている。イントラ用はDCT係数の低次の情報が人間の目に敏感で、高次の情報は視覚的マスキング効果により鈍感な事を利用してDC項と低次のAC項を細かく、高次のAC項を粗く量子化するように量子化係数がテーブル化されている。非イントラ用量子化マトリクスは動き補償予測画像と現画像の差信号データのDCT係数を取り扱う、このため一定の周波数依存性を持たない同一係数値の量子化係数テーブルが用意されている。

表4-1 イントラ用デフォルト量子化マトリクス

8	16	19	22	26	27	29	34
16	16	22	24	27	29	34	37
19	22	26	27	29	34	34	38
22	22	26	27	29	34	37	40
22	26	27	29	32	35	40	48
26	27	29	32	35	40	48	58
26	27	29	34	38	46	56	69
27	29	35	38	46	56	69	83

表4-2 インター用デフォルト量子化マトリクス

16	16	16	16	16	16	16	16
16	16	16	16	16	16	16	16
16	16	16	16	16	16	16	16
16	16	16	16	16	16	16	16
16	16	16	16	16	16	16	16
16	16	16	16	16	16	16	16
16	16	16	16	16	16	16	16
16	16	16	16	16	16	16	16

(b) 高域成分を低域より圧縮することによる得失

人間の目の性質から、低域は視覚的に敏感であるため画質を良くするため多くのビットを割り当て、視覚的に鈍感でしかもデータ量の多い高域を粗く量子化することにより、効率的な圧縮ができる、また、この反面、情報量を失った高域部分の影響で、デコードされた再生画像は、高域の量子化雑音の発生で高域のざわざわしたモスキートノイズを生ずる。

(c) 量子化マトリクスを適応化することによる得失

量子化の目的は、圧縮効果を得ることであるが、画像データブロックの性質により同様に圧縮しても劣化が目立つものと目立たないものがある、これをブロック毎に分析して適応的に量子化する。この場合の適応には周波数成分によるものと画像の局所的な特徴によるものがある。周波数成分による適応は前記のようにDCT係数の低次に多く高次に少なくビットを割り付ける量子化を行い、局所的特徴量に対しては、ブロック内の分散を検出して、変化の少ないところにビットを多く配分する。この結果雑音を人間が検知しにくい領域に集中させることにより画質を改善する。適応的量子化で全ての画像を完全に分析して適応化することは難しく、画像データブロックの特徴の分析にミスが生ずるとかえって劣化を招くことになる。

【量子化ステップの適応化】

GOP/マクロブロック単位で量子化ステップを切り替えることの得失

限られた符号量で高画質を得る為に画像の特徴をとらえ適応的に量子化する場合スライス単位の制御では範囲が広すぎて効率的な符号化ができない、そこで部分的な画像の変化に応じてマクロブロック単位で量子化ステップを切り替えている。また、全ての画像で適応化がうまくいくとは限らず、適応できない画像ではかえって画質劣化する場合も生ずる。

【デッドゾーン】

図4に示すように、どのような入力値に関わらず出力値がゼロとなるような量子化において、この入力値の範囲をデッドゾーンと呼ぶ。MPEGの量子化において、イントラに対してはデッドゾーンなし、インターに対してデッドゾーン付き量子化を行っている。インターのコーディングで小レベルの値を"0"にすることにより、"0"の連続する長さであるZero Run Lengthを長くとり符号化効率をあげることができる。

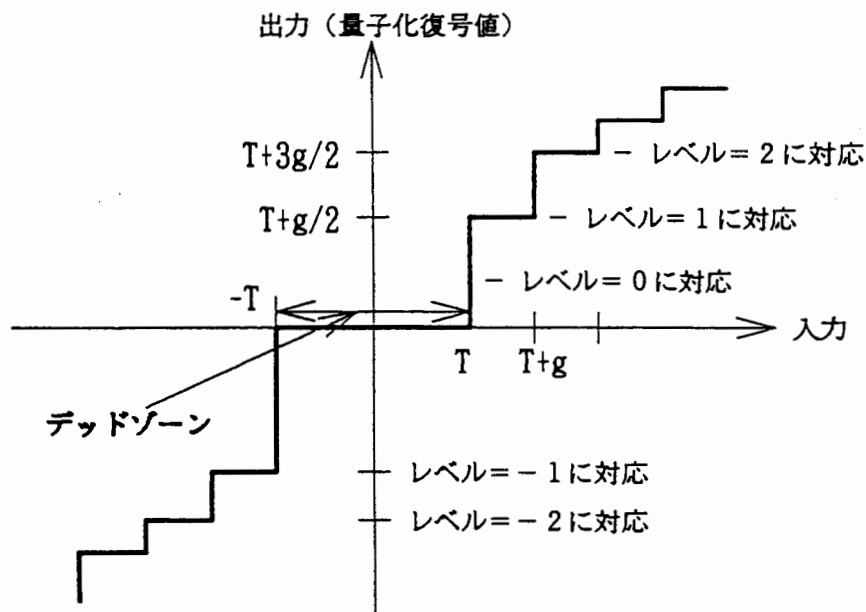


図4 デッドゾーン

【IDCTミスマッチ】

IDCT の計算方法の違いで、各社のデコーダは再生画像の正確な値が異なる可能性がある。このため、デコーダの再生画像と、エンコーダのローカル・デコーダとの違いが発生する。Pピクチャーでは、誤差がフレーム・メモリに累積されるので好ましくない。この基本的な原因は、IDCTが実数計算なのにデータは整数であるため計算誤差によって発生する。

例えば画素の理論値が" 整数+0.5" の時、標準が規定する精度がどんなに高めても丸めによって結果が1だけ違うことが避けられない。これを丸め法の規定によって避けることは可能であったが、ITU-T H. 261標準化前にすでにDCT、IDCTチップが市販されていたため、IDCTに与える数値をすべて奇数に制限することによって解決している。

【線形量子化と非線形量子化】

量子化器の設計において、線形・非線形のいずれを用いてエンコードするかについては、規定されていない。エンコード側では、量子化入力のどの範囲をどのレベルに対応させるかは、設計の自由となっている。デコード側については、相互接続性を補償するため、量子化特性に対する量子化復号値が定められている。

一般に、入力信号レベルの分布がある特定のパターンを持つ場合、非線形量子化を用いることにより、量子化効率の向上や復元画像の高画質化に役立つと考えられる。この考え方は、MPEGの音声符号化技術で積極的に使われている。音声信号レベル分布が指数関数的であるという特性から非線形量子化を用いている。画像に関しても、人間の知覚能力は変化の少ない平坦の所では敏感で、大きく変化するエッジのようなところでは鈍感なところから、粗密を組み合わせた非線形量子化が有効と思われるが、このような分布の特性についてまだ明確になっておらず、いずれを用いるかは設計側に任されている。即ち、MPEG1ではMQANTは線形量子化のみであったが、MPEG2では、表4-3にあるような非線形MQANTが追加され、ピクチャーレイヤで線形と非線形の量子化とを切り替えて使うことが可能である。

表4-3 量子化スケールコードと量子化スケール

quantizer_scale code	quantizer_scale	
	qscale_type=0	qscale_type=1
0	(forbidden)	(forbidden)
1	2	1
2	4	2
3	6	3
4	8	4
5	10	5
6	12	6
7	14	7
8	16	8
9	18	10
10	20	12
11	22	14
12	24	16
13	26	18
14	28	20
15	30	22
16	32	24
17	34	28
18	36	32
19	38	36
20	40	40
21	42	44
22	44	48
23	46	52
24	48	56
25	50	64
26	52	72
27	54	80
28	56	88
29	58	96
30	60	104
31	62	112

5. 動き検出

【ブロックマッチングを取る方法】

- 1) 差の絶対和 (MAE[Mean Absolute Error])
チェビシェフノルム (差の絶対値和) を最小にするマッチング方法
- 2) 2乗エラー (MSE[Mean Square Error])
2乗ノルム (差の2乗和) を最小にするマッチング方法

上記の2つの方法の性能はMSEがMAEよりわずかに勝っている。

実際の装置に於いてはMAEとMSEの性能差はほとんど無く、しかも計算量ではMAEの方がMSEより少ないので殆どMAEが採用されている。

【動きベクトルの検出方法】

- 1) フルサーチ
ある範囲の全ての可能な動きベクトルを全て調べ、最小の誤差を与えるベクトルを求める。

長所

単純な処理の為LSI化に適した方法である。

短所

計算量はマッチング窓の面積と探索範囲の面積の積になるので膨大な量になる。

2) Stepサーチ

ツリーサーチとも呼ばれる。

例えば、ある範囲の可能な動きベクトルを例えば4画素単位に最小誤差を与えるベクトルを求め、次に求められたベクトルを中心とした±4画素の範囲を2画素単位で最小誤差を与えるベクトルを求め、最後に求められたベクトルを中心とした±2画素の範囲を1画素単位で同様にベクトルを求めると言うように段階的にサーチする画素単位を細かくしていく方法である。

長所

1)の方法に比べて計算量が大幅に削減出来る。

半画素精度のベクトルを求める時には計算量が膨大となるので1画素→半画素、2画素→半画素、2画素→1画素→半画素と言うようなStepサーチの方法が用いられる。

3) 階層的動きベクトルサーチ

2)と同様に段階的にベクトルを求める方法であるが、マッチングを求める時に使うマッチング窓と探索範囲の画像に縮小画像を用いる方法である。例えば2)の例と同様の事を行う時、まず第一にマッチング窓と探索範囲の画像として縦横4分の1の縮小画像を用いてベクトルを求め、第二に縦横2分の1の縮小画像を用いてベクトルを求め、最後に原寸の画像を用いてベクトルを求める方法である。

長所

2)の方法より更に計算量を削減することが出来る。

4) テレスコピックサーチ

フレーム間隔が離れた時のベクトルサーチに用いる方法で、フレーム間隔 m の時、フレーム間隔 $m-1$ のベクトルを初期値としてフレーム間隔 m のベクトルを求める方法である。つまりフレーム間隔3のベクトルを求める時には、フレーム間隔1のベクトルを初期値としてフレーム間隔2のベクトルを求め、フレーム間隔3のベクトルはフレーム間隔2のベクトルを初期値として求める。

短所

Bピクチャを用いる時はエンコード処理順と対応が良くないので実際のハードウェアには適応できない。

5) ダイレクトフルサーチ

フレーム間隔が離れた時のベクトルサーチを行う時に、ベクトルサーチを行う探索範囲をフレーム間隔によって変えていく方法である。

具体的にはフレーム間隔あたりの探索範囲を $M \times N$ とした時に、フレーム間隔が2の時は $2M \times 2N$ 、フレーム間隔3の時 $3M \times 3N$ の探索範囲を設定しベクトルを求める事である。

短所

ハードウェアの規模が非常に大きくなる。

6. 動き補償

【フレーム予測／フィールド予測／デュアルプライム予測】

〔概要〕

符号化の単位により、フレーム構造とフィールド構造がある。フレーム構造では、フレーム／フィールド予測、デュアルプライム予測を備え、フィールド構造では、フィールド予測、16×8予測、デュアルプライム予測を備える。

フレーム予測

フレーム単位に動き補償を行う。順方向予測の参照画像は、直前のIまたはPフレームとなる。

フィールド予測

フィールド単位に動き補償を行う。順方向予測の参照画像は、直前のIまたはPの2フィールドとなる。

16×16画素のマクロブロックに対して、16×16画素単位に1ベクトルを用いるか、16×8画素単位に2ベクトルを用いるかを選択できる。

デュアルプライム予測

片方向予測においてのみ用いられ、参照する両フィールドからの予測を平均する。1つのベクトル(MV)を用い、参照フィールドと予測フィールドの時間間隔に合わせて、ベクトルを伸張する。伸張ベクトルは、修正ベクトル(dmv)により補正がなされる(図6-1参照)。

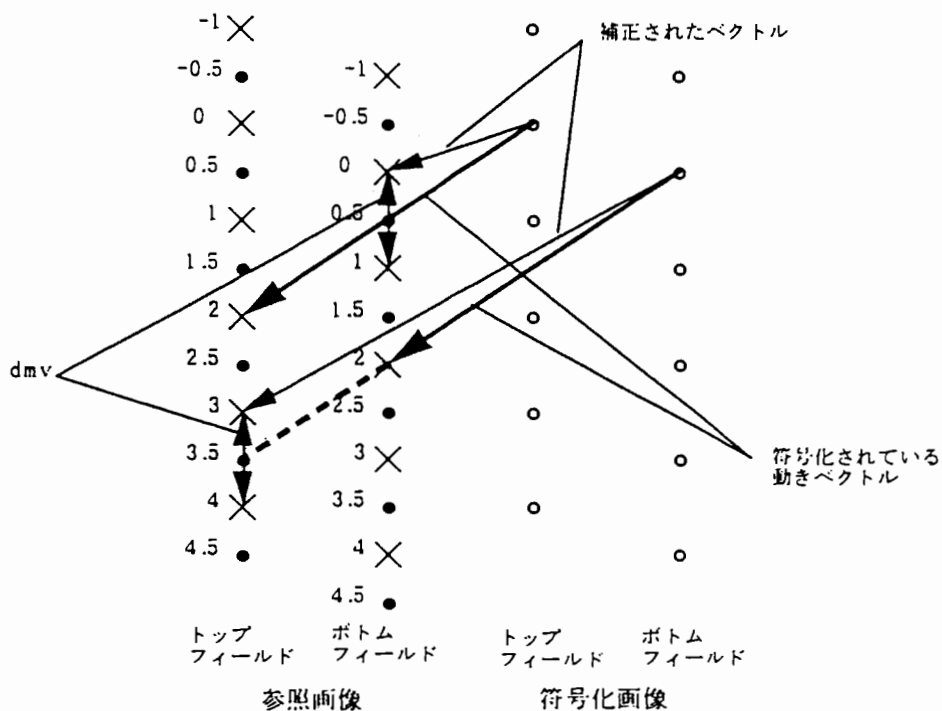


図6-1 デュアルプライム予測

[特徴]

フレーム予測

- ・静止面の細かいテクスチャに適する。

フィールド予測

- ・動きの激しいシーケンスに適する。

デュアルプライム予測

- ・インタレース画像での予測精度向上。
- ・エンコーダの負担が大きく、リアルタイムエンコーダでの使用は疑問視されている。

【片方向予測／双方向予測】

[概要]

片方向予測

- ・過去に符号化した画像から予測する方式（順方向予測）であり、ITU-T H.261において採用されている。
- ・順方向予測とは別に、未来から予測する逆方向予測もある。

双方向予測

- ・順方向予測と逆方向予測を対応画素にて平均して用いる方式であり、MPEG特有のものである。
- ・MPEGでは、イントラのみのIピクチャ、片方向予測のみのPピクチャ、両方向予測を含むBピクチャで、GOPを構成する。

[特徴]

片方向予測

- ・ハードウェア化が容易。
- ・遅延時間が少なく、通信に向いている。

双方向予測

- ・予測効率が良い。
- ・画面の処理順が、原画面の順番と異なる。
- ・エンコーダ、デコーダにより、I/Pピクチャの周期だけ遅延。
- ・前後予測に独立の動きベクトルが必要であり、符号化量が増加。

【1/2画素精度動き補償】

【概要】

- ・動き補償を、0.5画素精度で行う。
- ・予測画素位置が2画素の間であれば、2画素値の平均を予測値とする。また、4画素の中心であれば、4画素値の平均とする。

【特徴】

- ・予測精度の向上。
- ・上記の画素平均により、空間フィルタの効果がある。
ITU-T H.261での2次元ループ内フィルタに相当する役割を果たしている。
- ・フレーム画像の垂直成分に対しては、時間フィルタの効果がある。

7. 可変長符号化

【2次元ハフマン符号化】

【概要】

- ・量子化された2次元DCT係数をジグザグスキャンし、連続するゼロの個数と非ゼロ値とを一括してVLC (Variable Length Code) テーブルに従って符号化する。
- ・ブロックの終了はVLCテーブルに含まれるEOB (End Of Block) コードで識別する。EOBはブロック内の2次元DCT係数すべてを符号化した場合でも必ず付加する。
- ・符号長が長くなるものについてはVLCテーブルに含まれるエスケープコードに続けて、係数値そのものを送る。
- ・イントラピクチャの2次元DCT係数のDC項のみは他の項と別の符号化が用いられる。
- ・輝度信号と色差信号は別々に符号化される。

長所

(1次元ハフマン、固定長符号化と比較)

- ・1次元ハフマン、固定長符号化と比べ符号化効率が高いのが特徴。
- ・EOBを用いることにより、符号化の打ち切りが可能で、DCT係数が低次項に集中している場合は符号化効率が良くなる。

短所

- ・復号化はゼロランと非ゼロ値を分離しそれぞれ処理をするため複雑になる。
 - *ビットバイビット復号法ではハードウェア量が小さいのが長所であるが、ビットレートの上昇に従って、フィードバックループがネックになるため比較的低い周波数で動作限界を迎える。
 - *パレルシフトを用いるパレル復号法では動作周波数限界は高いがハードウェア量はかなり多くなる。
- ・ワード同期とワード復号をその都度行っていくためビットエラーによる影響を受けやすい。誤りが生ずると連鎖する。
- ・統計的に符号表が作成されているため、分布から大幅にはずれた入力に対しては符号量が増大する危険性を有する。
- ・ブロックすべての係数を符号化しても更にEOBを付加するため、冗長になる。

【動きベクトル符号化】

動きベクトルは±16までを表すVLCとフレーム間隔を表すビットで構成され1つの成分の差分が表現される。

【変換係数 (DC, AC) の符号化】

- ・イントラピクチャのDC項は4つの輝度ブロックの差分のサイズをVLCで記録し、それに続けて値の有効桁のみを表現する。

- ・色差と輝度では異なったVLCが用意されている。
- ・イントラピクチャのDC項以外はDCT変換係数は同一に扱われ、2次元ハフマン符号で表現される。

【VLCテーブル (Variable Length Code)】

- ・連続するゼロの個数と非ゼロ値によって定まる符号表で113個用意されている。
最短符号長は2ビット、最長符号長は17ビットである。
- ・符号表にあるVLCコードの最終ビットは非ゼロ値の極性に相当する。
“0”は正、“1”は負に相当する。
- ・EOBも符号表の1つの符号として割り当てられている。(10)
- ・符号表で記述されていないものについてはVLC中のエスケープコード(0000 01)に続けて、6ビットで連続するゼロの個数を表し、その後に12ビットの固定長符号で非ゼロ値を表す。

8. レートコントロール

【概要】

MPEG2 TM で定めるレートコントロール方法は、量子化係数をマクロブロック単位で制御するものであり、以下の3stepから実現される。

- step1: GOP内の残り符号量計算によるピクチャごとの符号量推定
- step2: ピクチャ種類別のバッファ残量計算による量子化係数制御
- step3: マクロブロックの輝度の Activity値による量子化係数制御

step1: Target Bit Allocation

ピクチャごとに符号化が終了した時点で、GOP内の次に符号化する同タイプ(I/P/B)のピクチャの推定符号量 $T_i/T_p/T_b$ を算出する。

まず、現在のピクチャの複雑さを示すパラメータ "Global Complexity" $X_i/X_p/X_b$ を求める。

$$X_i = S_i \times \bar{Q}_i, \quad X_p = S_p \times \bar{Q}_p, \quad X_b = S_b \times \bar{Q}_b$$

- ・ $S_i/S_p/S_b$: 現在のピクチャの符号量(bits)
- ・ $\bar{Q}_i/\bar{Q}_p/\bar{Q}_b$: 現在のピクチャで使用した量子化パラメータの平均値
- ・ $X_i/X_p/X_b$ の初期値は以下のように計算する。 bit_rate の単位は bit/s
$$X_{i\text{init}} = 160 \times bit_rate / 115$$
$$X_{p\text{init}} = 60 \times bit_rate / 115$$
$$X_{b\text{init}} = 42 \times bit_rate / 115$$

あらかじめ、1GOPの割り当てビット数を想定しておき、GOP中の残りビット数をピクチャタイプ(I/P/B)換算した残りピクチャ数で割ることによって、次に符号化するピクチャの推定符号量 $T_i/T_b/T_p$ を求める。

$$T_i = \max\left\{\frac{R}{1 + \frac{N_p X_p}{X_i K_p} + \frac{N_b X_b}{X_i K_b}}, \frac{bit_rate}{8 \times picture_rate}\right\}$$

$$T_p = \max\left\{\frac{R}{N_p + \frac{N_b K_p X_b}{K_b X_p}}, \frac{bit_rate}{8 \times picture_rate}\right\}$$

$$T_b = \max\left\{\frac{R}{N_b + \frac{N_p K_b X_p}{K_p X_b}}, \frac{bit_rate}{8 \times picture_rate}\right\}$$

- ・ $K_b = 1.0$, $K_p = 1.4$ (量子化マトリックスに依存する定数)
- ・ R は GOP 中の残りビット量
- ・ N_p/N_b は GOP 中の残り(未符号化) P/B ピクチャの数

step2: Rate Control

ピクチャタイプ(I/P/B)別の仮想バッファを使って、マクロブロック単位で量子化パラメータの基準値を算出する。

マクロブロック j の符号化前に、I/P/B各仮想バッファ内のフルネス $d_i_j / d_p_j / d_b_j$ を計算する。

$$d_i_j = d_i0 + B_j(j-1) - \frac{T_i \times (j-1)}{MB_cnt}$$

$$d_p_j = d_p0 + B_j(j-1) - \frac{T_p \times (j-1)}{MB_cnt}$$

$$d_b_j = d_b0 + B_j(j-1) - \frac{T_b \times (j-1)}{MB_cnt}$$

- ・ $d_i0/d_p0/d_b0$ は各仮想バッファの初期フルネス。

$$d_i0 = 10 \times r / 31$$

$$d_p0 = K_p \times d_i0$$

$$d_b0 = K_b \times d_i0$$

- ・ B_j はピクチャ開始からマクロブロック j までに発生した符号量(bits)。
- ・ MB_cnt はピクチャ内のマクロブロック数。

仮想バッファフルネスから量子化パラメータ基準値 Q_j を計算する。

$$Q_j = d_j \times 31 / r$$

ここで、reaction parameter r は次式で与えられる。

$$r = 2 \times \text{bit_rate} / \text{picture_rate}$$

step3: Adaptive Quantization

マクロブロックごとの空間Activity値によって最終的な量子化パラメータ値を算出する。

まず 8x8画素のブロック毎に、原画像の輝度値 P_k の平均値との差の自乗和 var_sblk を求める。

$$P_mean = \frac{1}{64} \sum_{k=1}^{64} P_k$$

$$\text{var_sblk} = \frac{1}{64} \sum_{k=1}^{64} (P_k - P_mean)^2$$

次に、空間的 Activity値 act_j と、その正規化値 N_act_j を求める。 avg_act は直前に符号化したピクチャで用いた act_j の平均値である。

$$act_j = 1 + \min_{sblk=I, B}(\text{var_sblk})$$

$$N_act_j = \frac{2 \times act_j + avg_act}{act_j + 2 \times avg_act}$$

正規化アクティビティ値 N_act_j と step2で求めた Q_j 基準値から量子化パラメータ $mquant_j$ を算出する。

$$mquant_j = Q_j \times N_act_j$$

以上の処理で量子化パラメータ $mquant_j$ を決定する。

【TM方式の長所】

- (1) 符号量制御がGOP毎に完結している。
- (2) ピクチャ毎の目標符号量を動的に制御するので、柔軟な対応が可能。
- (3) 視覚特性を利用した Activity制御によって、SNRは低下するが視覚上画質の向上が得られる。

【TM方式の短所】

- (1) 時間的に前の画像の性質から後の画像の符号量を推定するため、シーンチェンジを挟む場合や、シーケンス開始時には有効でない。
- (2) VBV (Video Buffer Verifier) のオーバーフロー、アンダーフローが保証されていないので、エンコーダが別途制御する必要がある。
- (3) Activity値をフレーム構造のブロックから得ているので、動きの大きいインタレース画像をフィールド処理する場合に矛盾が生じる。

9. 前処理／後処理

9. 1 4:2:0フォーマット

TV画像のコンポーネント信号のデジタル符号化規格としてITU-R Rec. 601がある。これは輝度信号Yと2つの色差信号Cb、Crからなり、525システムでは図9-1に示すようなデータの並び方をしている。これからY、Cb、Crを拾い集めてみると、輝度信号Yは1走査線当たりの有効サンプル数が720となり、また2つの色差信号Cb、Crは横方向にサブサンプリングされて1走査線当たりの有効サンプル数が360となる。通常これを「4:2:2」と呼ぶが、MPEG2のメインプロファイルおよびシンプルプロファイルでは「4:2:2」の色差信号をさらに縦方向に1/2にサブサンプリングした「4:2:0」と呼ばれる画像形式で処理を行う。（「4:2:2」はハイプロファイルでのみサポートされている。）

人間の視覚特性は色差信号よりも輝度信号に対してより敏感であることから、「4:2:0」を用いることは、データ量を25%削減すると同時に色差信号よりも輝度信号により多くの情報量を割り当てることとなるため、視覚上の影響をほとんど与えることなく圧縮効率を上げることができる。

ここで「4:2:2」から「4:2:0」へ符号化形式を変更する処理を前処理、「4:2:0」から「4:2:2」へ戻す処理を後処理というが、これらの処理に関しては基本的にMPEGの標準化外の技術であるため、フィルタの特性などについては各エンコーダ／デコーダメーカーの工夫のしどころである。MPEG2において、TMの段階では前処理として色差信号の最初のフィールドは7タップフィルタ（ $[-29, 0, 88, 138, 88, 0, -29]/256$ ）で縦にサブサンプリングし、次のフィールドは4タップフィルタ（ $[1, 7, 7, 1]/16$ ）で縦にサブサンプリングしていた。しかしながら1994年3月のDISの段階では、色差サンプルの位置などについて図9-2のように変更された。

また後処理の一つとして、カメラからのランダムノイズ除去や、照明によるフリッカ除去などが行われることもある。

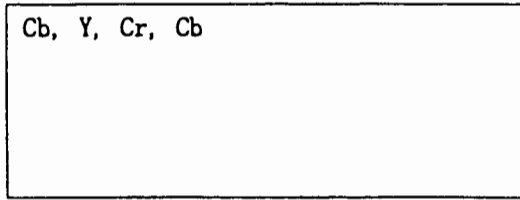
長所

輝度信号に割り当てる情報量が同じである場合、4:2:0は4:2:2に比べてビットレートを25%削減できる。

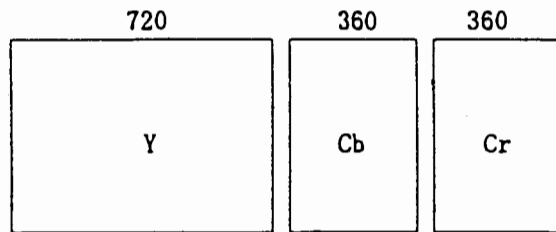
短所

トータルのビットレートが同じである場合、4:2:2に比べて4:2:0では色の解像度が劣化する。

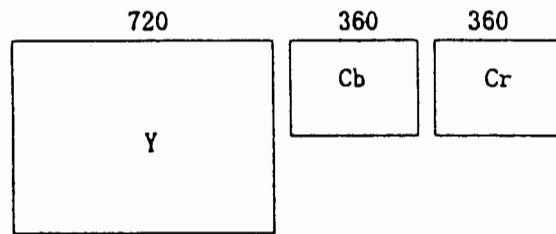
1440画素



(1) ITU-R Rec. 601形式

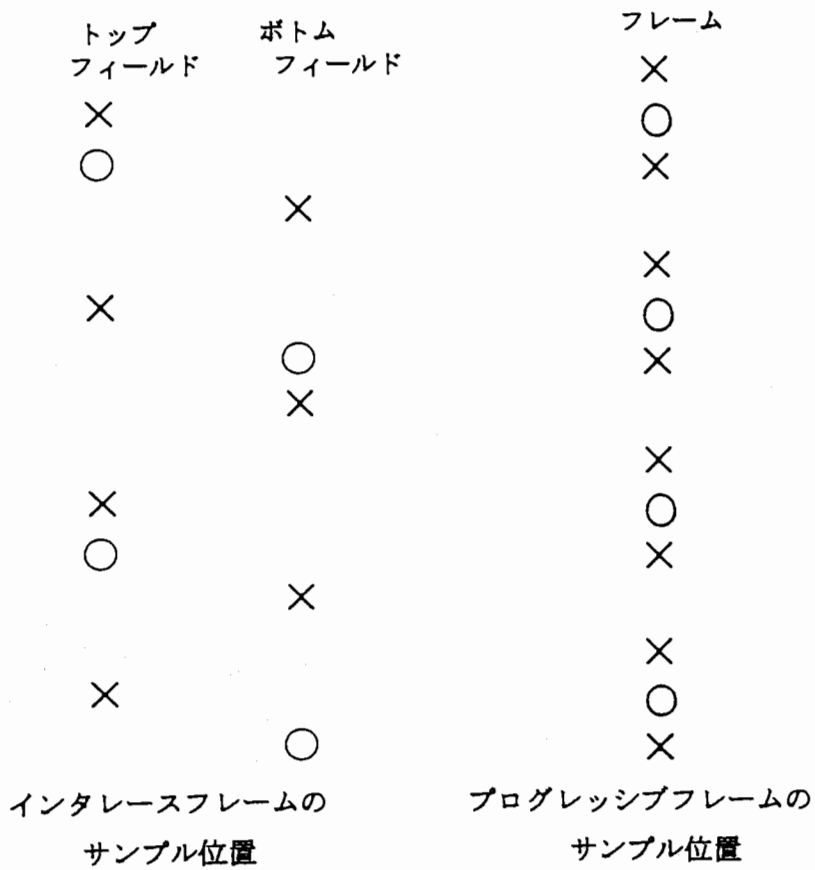


(2) 4:2:2形式



(3) 4:2:0形式

図9-1 画像フォーマット



×	×	×	×	×	×	×	×
○		○		○		○	
×	×	×	×	×	×	×	×
×	×	×	×	×	×	×	×
×	×	×	×	×	×	×	×
○		○		○		○	
×	×	×	×	×	×	×	×
×	×	×	×	×	×	×	×

- × 輝度 (Y) サンプル位置
- 色差 (Cb, Cr) サンプル位置

図9-2 輝度および色差サンプルの位置

9. 2 3/2プルダウン

映画のデータ（～24fpsのフィルム）をテレビ（～30fpsのビデオ）で放映する時に使用される。具体的には、予め映画のデータをプログレッシブ画像のシーケンスとして符号化して伝送する。映画の各画面はビデオのフィールド周期の2周期か3周期かのどちらかで表示されるように独立に決められる。3/2プルダウンに関する情報は、Picture Coding Extensionのtop_field_firstとrepeat_fieldの2つのフラグに記述され、デコード時に適切なタイミングで表示される。

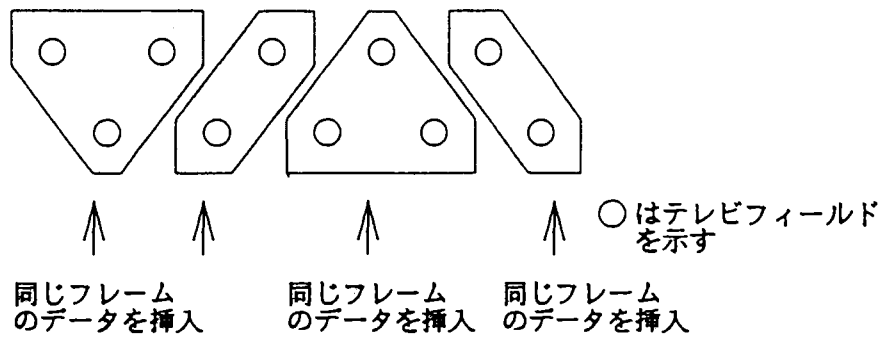


図9-3 3/2プルダウンのイメージ[14]

9.3 パン/スキャン

例えば、HDTVの16:9の符号化画像のデータをSDTVの4:3の画面上に表示するために、16:9の画像上に4:3のアスペクト比の表示画面をウィンドウの形で設定することができる。これはちょうど16:9の画面全体にわたって4:3の画面を持ったカメラをパン（上下左右に動かす）したような効果を与える。

パン/スキャンの実現は、sequence display extensionで表示する長方形領域（上の例で言うと4:3の表示画面）の大きさを規定し、picture display extensionでその表示長方形の位置を決めることで可能となる。下にパン/スキャンの概略を図示する。表示画面の位置は1/16画素精度で指定できる。

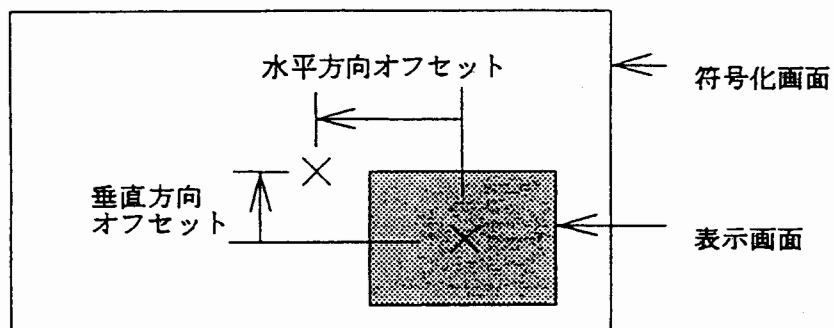


図9-4 パン/スキャンの概要

10. スケーラビリティ (階層構造)

【概要】

一般にデジタル信号では、伝送する情報をどのような配列で送出するか任意に操作可能であり、その情報の伝送を階層構造とすることが可能である。放送応用としては、送出される信号を階層構造とすることにより、受信者は受信状態や受信機の性能に応じて異なるサービス品質を得ることができる。

スケーラビリティにはさまざまなタイプのものを考えることができるが、ここではMPEG2 DIS [13]にて規定されている空間(Spatial)およびSNRスケーラビリティを中心に解説する。

【空間スケーラビリティ】

空間解像度の異なる複数の階層を持つ。エンコーダは、ダウンサンプルされた画像に対して符号化を行った基本レイヤ(Base Layer)、およびその信号を復号後アップサンプルした画像と原画との誤差分をさらに符号化することで得られる補強レイヤ(Enhancement Layer)の2種類のストリームを出力する。応用例として、HDTVとSDTVとのコンパチビリティを実現するためのエンコーダのブロック図を図10-1^[7]に示す。

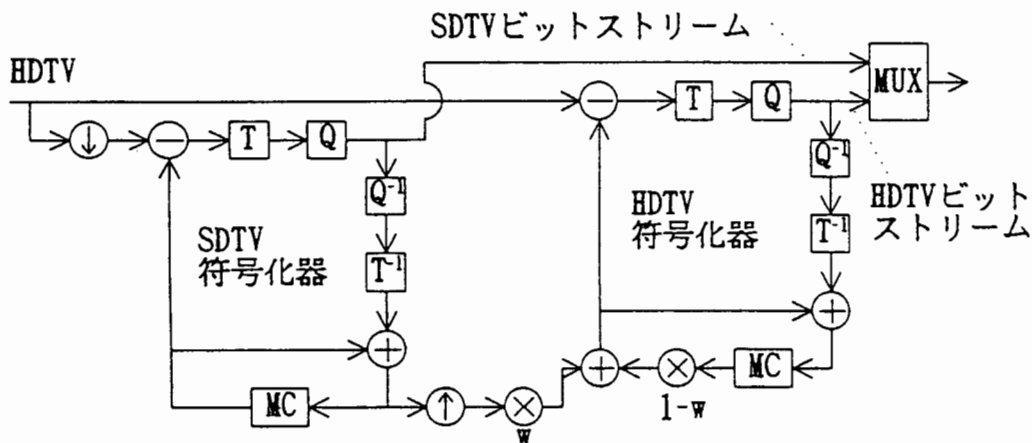


図10-1 HDTV-SDTV空間スケーラビリティのエンコーダ構成

【SNRスケーラビリティ】

解像度の等しい複数の階層を持つ。エンコーダは、粗い量子化により得られる基本レイヤと共に、その量子化/逆量子化に伴うDCT領域での誤差分をさらに細かく量子化することにより得られる補強レイヤの2種類のストリームを出力する。エンコーダのブロック図を図10-2^[7]に示す。応用例としては、基本レイヤ部の誤り耐性を強くしておくことで実現できるグレースフル・デグラデーション(Graceful Degradation)がある。

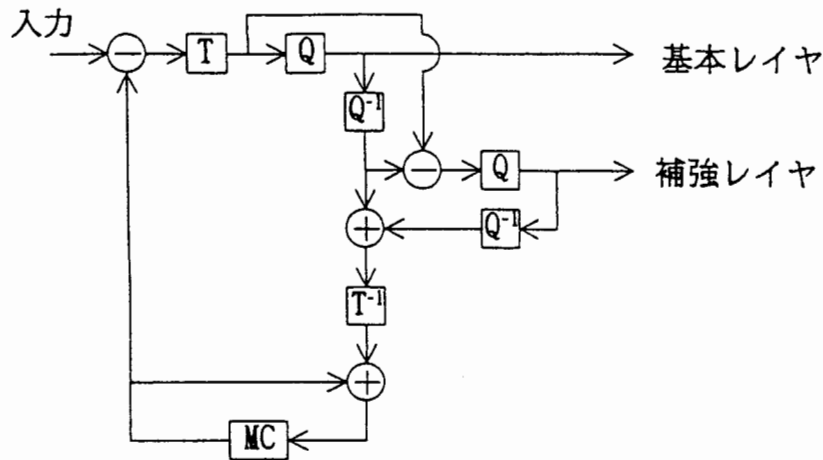


図10-2 SNRスケーラビリティのエンコーダ構成

【その他】

時間(Temporal)スケーラビリティ

順次走査フォーマットと飛越し走査フォーマットとのコンパチビリティを実現するために用いられる。MPEG2 DISに記述されているが、現段階ではどのプロファイルにも属していない。

データパーティショニング(Data Partitioning)

例えばATMセル廃棄対策に用いられ得るもので、符号化後のマクロブロック係数を低域と高域とに分割して伝送する。上と同様MPEG2 DIS^[13]に記述されているが、どのプロファイルにも属していない。

周波数(Frequency)スケーラビリティ

異なる解像度間のスケーラビリティを実現するため、(空間領域でなく) DCT領域にて階層符号化を行うもの。MPEG2では採用されず。

クロマ(Chroma)スケーラビリティ

異なるクロマフォーマット(4:2:0、4:2:2、4:4:4)間のコンパチビリティ実現を目的とする。MPEG2では採用されず、この機能はクロマサイマルキャストによって実現されることになった。

10ビットスケーラビリティ

8ビットと10ビットのコンパチビリティ実現を目的とする。現段階のMPEG2 DISでは規定されていない。

【非階層符号化との比較】

- 長所
- ・階層化サービスを提供できる
 - ・サイマルキャストに比べて効率が良い（ただしクロマスケラビリティではその有効性が認められなかった）
- 短所
- ・オーバーヘッドの増大に伴う符号化効率の低下
 - ・ハードウェア規模、複雑さが増大
 - ・複数レイヤ間のレート制御が複雑になる（エンコーダ）
 - ・同期の難しさが増大（デコーダ）

【他のスケラビリティとの比較】

空間スケラビリティ

- 長所
- ・任意の解像度比に対応可能
- 短所
- ・階層数に応じた符号化器が必要なため、ハードウェア規模、複雑さの点で不利
 - ・アップサンプリング・ダウンサンプリングフィルタのマッチングがとれないと画質劣化を引き起こす

SNRスケラビリティ

- 長所
- ・DCT符号化器が1つで済むため、ハードウェア規模、複雑さの点で有利
 - ・符号化効率が比較的高い
- 短所
- ・同じ解像度間の階層化のみにしか利用できない

1 1. ProfileとLevel

【概要】

MPEG2では、MPEG1並の小さい画素サイズからHDTVまでの幅広いアプリケーションに適応できるように”Level”と”Profile”という概念が用いられている。表11-1はMPEG2の標準仕様を示し、11個のコンFORMANCE・ポイント（表の○印）が認められている。Level（表の縦軸）では、画素サイズや目標符号化レートの上限值を4段階に分けており、Profile（表の横軸）では、符号化ツールの複雑さや階層処理の有無などにより5種類に分けている。縦軸は上に行くほど解像度が向上し、横軸は右に行くほど圧縮技術が高度になり、圧縮率が高くなる。つまり右上に行くほど画質が上がり、使用する技術が高度になる。互換性については、Profileの広範囲方向（表では右方向）、およびLevelが高くなる方向（表では上方向）が、狭いProfile、および低いLevelのビットストリームを解読できる上位互換を採用している。表11-1にMPEG2の標準使用を示す。

表11-1 MPEG2の標準仕様

Profile Level	Simple	Main	SNR Scalable	Spatial Scalable	High
High		○ *2			○
High-1440		○		○ *6	○
Main	○ *1	○ *3	○ *5		○
Low		○ *4	○		

*1: デジタル伝送のケーブルTV

*2: 米国のATV

*3: デジタル・ビデオディスク、衛星放送のDirecTV

*4: MPEG1相当

*5: 欧州の地上波デジタルTV

*6: 欧州のHDTV

以下に、LevelとProfileについての概要を示す。

【Levelについて】

Levelは画素数、フレーム数、ビットレート、動きベクトル範囲などが定義されている。表11-2に各Levelの最大画素サイズの定義を示す。

表11-2 Levelの定義

Levelの名称				備考
	最大水平画素数 (samples/line)	最大走査線数 (lines/frame)	最大フレームレート (frames/sec)	
High	1920	1152	60	HDTV相当
High-1440	1440	1152	60	HDTV相当
Main	720	576	30	ITU-R Rec. 601相当
Low	352	288	30	MPEG1/ITU-T H. 261 CIF相当

Low Levelは352画素×288ライン以下でMPEG1相当、Main Levelは720画素×576ライン以下でITU-R Rec. 601相当、High-1440は1440画素×1152ライン以下、Highでは1920画素×1152ラインでHDTV相当の画像に適応可能である。

【Profileについて】

Profileは符号化構造のサブセットを定義するもので、クロマフォーマット、スケーラビリティ機能の有無、Bピクチャーの有無などが定義されている。大きく分けるとスケーラビリティのないSimple Profile、Main Profileとスケーラビリティを使用できるSNR Scalable、Spatial Scalable、High Scalableの2つに大別できる。表11-3にProfileの定義を示す。

表11-3 Profileの定義

Profileの名称	クロマフォーマット	Spatial S.	SNR S.	Bピクチャ
Simple	4:2:0	×	×	×
Main	4:2:0	×	×	○
SNR Scalable	4:2:0	×	○	○
Spatial Scalable	4:2:0	○	○	○
High	4:2:0, 4:2:2	○	○	○

Simple ProfileはBピクチャーを使用しない構造なので、低遅延やハードウェアの簡易化を図ることができる。

Main ProfileはITU-R Rec. 601画像を、両方向予測を用いて符号化することができ、フィールド構造に対応した点がMPEG1との相違点である。

SNR Scalableでは、高SNRと低SNRの2階層の画像を同時に送ることが可能である。

Spatial Scalableでは、高解像度と低解像度の2階層の画像を同時に送ることが可能である。

High Scalableでは、4:2:2のクロマフォーマットに対応でき色の高解像度化が可能で、Main Level以上で使用可能である。

【Main Profile @ Main Levelについて】

表11-1における*3がMain Profile @ Main Level (メイン・プロファイル・メインレベル) と呼ばれる標準セットで、放送、通信、コンピューターなどのマルチメディアにおいて横断的に最も活用範囲が広いモードであり、画質的には、ITU-R Rec. 601に相当する。表11-4にこの仕様を示す。

表11-4 Main Profile @ Main Levelの仕様

項目	内容
1. 画像フォーマット	ITU-R Rec. 601(720×480×29.97Hz, 720×576×25Hz) 以下
2. ビットレート	15Mbps
3. クロマフォーマット	4 : 2 : 0
4. ビクチャタイプ	Iピクチャ、Bピクチャ、Pピクチャー
5. 符号化画像単位	フレーム構造およびフィールド構造
6. 動き補償予測	
フレーム構造	16×16フレーム予測、16×8フィールド予測、 Dual-Prime(IピクチャとPピクチャの場合)
フィールド構造	16×16フィールド予測、16×8フィールド予測、 Dual-Prime(IピクチャとPピクチャの場合)
7. 動き補償精度	垂直方向[-127.5, +128]、0.5画素精度、 水平方向[704.0, +704.0]
8. バッファサイズ	1.835008Mbits以下
9. 互換性	MPEG1フォワード互換
10. イントラDC精度	8、9、10bits
11. イントラVLC	MPEG1/NEW Table(ビクチャレイトで切り替え可能)
12. DCT係数スキャン	MPEG1/NEW Scan(ビクチャレイトで切り替え可能)
13. エラー耐性	イントラマクロブロックでは動ベクトルによるコンソールメント
14. VBR動作	含まれる

12. シンタックス

デジタル映像をMPEGに準拠して符号化したデータ列すなわちビットストリームは符号化器から復号器へ間違いなく伝達され解釈されなければならない。そのためにMPEG1では ISO/IEC 11172-2、MPEG2では ISO/IEC 13818-2 の国際標準規格（案）に制定されている規則（シンタックス）に則ってビットストリームが構成され出力されなければならない。本節ではその概略を説明する。

図12にMPEG1と上位互換のMPEG2のビットストリームの構造の概略図を示すが、その基本はヘッダとデータにあり、Committee Draft にはC言語的に記載されている。ヘッダにはまず '0x000001□□' (HEX) の32ビットのスタート・コードが入っており、□□に入る各識別コードは表12-1の通りである。この識別によりそれ以後のデータが何についてのものであるか、どのように解釈するかがわかるのである。このためヘッダはデータとして発生しないユニークなコードになっている。

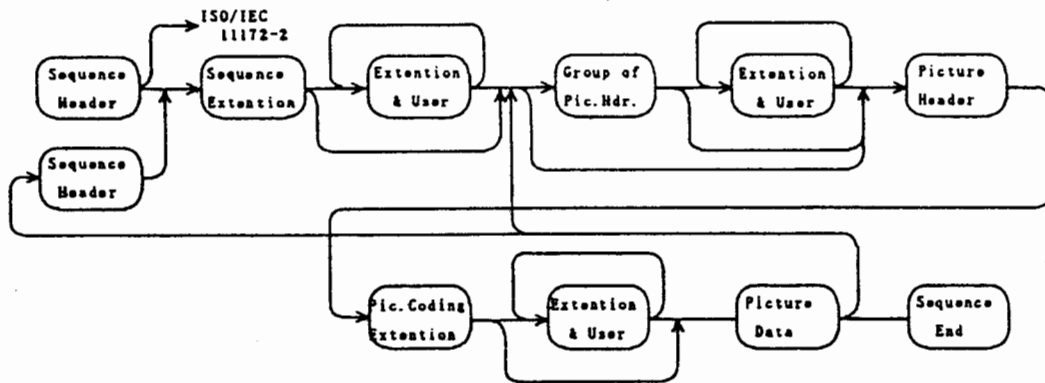


図12 ビットストリームの構造

表12-1 start code の値

name	HEX
picture_start_code	00
slice_start_code	01~AF
reserved	B0
reserved	B1
user_data_start_code	B2
sequence_header_code	B3
sequence_error_code	B4
extension_start_code	B5
reserved	B6
sequence_end_code	B7
group_start_code	B8
system_start_code	B9~FF
(system_start_codeの詳細は13818-1に定義)	

MPEGビデオ・デコーダはデータ列を受信すると、まずビデオ・シーケンス層として32ビットの sequence_header_code (表12-1参照)を探し出す。データ列にこのスタート・コードが存在することはすなわちデータ列がMPEGのビットストリーム(MPEG1とMPEG2で sequence_header_code は共通)であることを意味する。sequence_header()の中には符号化された映像の水平と垂直の解像度、アスペクト比、フレームレート、ビットレート等が記述されている。またデフォルト以外の量子化マトリックスを使用した場合は、そのマトリックスが記述されている。ユーザ・データも user_data_start_code を用いてこの中に記述することができる。

sequence_header()の後に続く32ビットが extension_start_code であればMPEG2のビットストリーム、そうでなければMPEG1のビットストリームと判断して以降の復号の処理が進められる。MPEG2では extension_start_code の次の extension_start_code_identifier (表12-2参照)を見て sequence_extension()が解読される。この中にはプロフィールとレベル、プログレッシブ・スキャン、色差形式、水平垂直サイズの拡張、ビットレートの拡張、低遅延などが記述されている。また extension_and_user_data(0)にこのビデオ・シーケンスを通しての extension_data(0) (ディスプレイ拡張やスケーラブル拡張)や user_data()を入れることができる。

ビデオ・シーケンス層に続いてGOP層のヘッダ group_of_pictures_header()が記述される。この中には group_start_code に続いてGOPの先頭のピクチャの time_code 及びこのGOPの closed_gop, broken_link が記述されている。MPEG2では group_of_pictures_header()を省略することができる。すなわちビデオ・シーケンス層に続いてピクチャ層のヘッダを記述することができる。また group_of_pictures_header()を記述した場合、その後には extension_and_user_data(1)を記述することができる。

表12-2 extension_start_code_identifier

extension_start_code _identifier	name
0000	reserved
0001	Sequence Extension
0010	Sequence Display
0011	Quant Matrix
0100	reserved
0101	Sequence Scalable
0110	reserved
0111	Picture Pan-Scan
1000	Picture Coding
1001	Picture Spatial Scalable
1010	Picture Temporal Scalable
1011	reserved
1100	reserved
...	...
1111	reserved

ピクチャ層には、

```

picture_header()
picture_coding_extension()
extension_and_user_data(2)
picture_data()

```

が記述される。picture_header() には、picture_start_code に続いてテンポラル・リファレンス、符号化タイプ、VBVディレイが記述されている。PおよびBピクチャではこの後に

```

full_pel_forward_vector
forward_f_code
full_pel_backward_vector
backward_f_code

```

を記述するが、MPEG2では、vector=0 f_code=7 で固定である。次に extra_bit_picture が1ビット用意されており、この値が“1”の時に後ろに8ビットの extra_information_picture が追加できる。しかし Committee Draft では中身は未定義となっている。picture_header() に続いて picture_coding_extension() の記述が必要である。この中にはMPEG2としてそのピクチャを復号するのに必要な情報、例えば水平垂直の f_code や picture_structure などが記述されている。ピクチャ層の extension_and_user_data(2) には量子化拡張、ディスプレイ拡張、時間拡張、

空間拡張が `extension_start_code_identifier` で識別されて記述される。`picture_data()` にはスライス層のデータが格納される。

スライス層ではまず `slice_start_code` が記述されているが上位24ビットはスタート・コードの '0x000001' (HEX) であり、下位8ビットが '00' ~ 'AF' (HEX) の値である。この下位8ビットは `slice_vertical_position` を表している。2800ライン (175スライス) を超える映像に対しては、`slice_vertical_position_extension` が用意されている。また詳細は説明しないがビデオ・シーケンス層にスケーラブル拡張がある場合で `scalable_mode` が `data partitioning` であればここに `priority_breakpoint` が記述される。`quantiser_scale_code` はこの後に記述される。これと先の `q_scale_type` と組み合わせて `quantiser_scale` が求められる。次に `marker_bit` があり、値が "1" のとき `intra_slice` 他の指定ができる。スライス層でも `extra_bit_slice` が1ビット用意されており、この値が "1" の時に後ろに8ビットの `extra_information_slice` が追加できる。しかし Committee Draft では中身は未定義となっている。スライスの終了には '000 0000 0000 00 00 0000 0000' (BIN) を挿入する。

マクロ・ブロック層の先頭には `macroblock_escape` として11ビットの '0000 0001 000' (BIN) が記述される。続いて `macroblock_address_increment` が 1 ~ 11ビットの可変長符号で記述される。この可変長符号の示す値は通常 "1" であるが `skipped_macroblock` が発生すると "2" 以上の値を取る。`macroblock_modes()` にはマクロブロックの符号化タイプ他が記述される。スライス層で記述されている値と異なる `quantiser_scale_code` を用いる場合はその次に記述できる。また符号化タイプ別に必要となる動きベクトルの記述が続く。

ブロック層ではDCT係数をスキヤンの順に可変長符号化して記述する。

ビデオ・シーケンス層は `sequence_end_code` が受信されるまで続く。また `sequence_end_code` 受信後も `sequence_header_code` が受信されれば復号が再開する。

なお、MPEG2では、MPEG1 (ISO/IEC 11172-2) およびITU-T H. 261とのコンパチビリティがある。すなわち "sequence_extension" のないMPEG2ビットストリームがMPEG1との forwardコンパチビリティとなり、スペシャル・スケーラビリティを用いれば下位層でMPEG1のbackwardコンパチビリティとなる。またH. 261に対して forwardコンパチビリティについては Committee Draft ではどのプロファイルでもサポートされていないが、backwardコンパチビリティに対しては簡単な方法としてサイマルキャストがある。

1 3. エラー対策（耐性）

【概要】

- ・ 伝送路のエラー対策としては、一般に誤り訂正符号を用いることが多いが、その方式等についてはMPEGの規定外となっている。基本的には、MPEGで規定するパケットより下位のレイヤで、システム毎に独自の誤り訂正符号を付加するなどのエラー対策を行うこととなる。
- ・ 上述の誤り訂正符号とは別に、MPEGの規定中でエラー対策に関するものは、以下の2項目。

1. トランスポートストリーム

- ・ MPEG-SYSTEMで規定されるデータストリームの伝送形態の1つで、エラーのある伝送路向けのもの。
- ・ 可変長のPES (Packetized Elementary Stream) を固定長のTransport packet (188 Byte、以下TPと略記) 化して伝送する。
- ・ TPの固定長ヘッダ部 (Transport Header = 32bits) には、このTPにエラーがあるか否かを示す1ビットのフラグ (transport_error_indicator) 領域が確保されている。このフラグを書き込むためには、TPの外側で誤りの検出 (または訂正) を行なう必要があるが、その方式はユーザーあるいはシステム毎に独自に決定すればよいことになっている。
- ・ 米国のATVもこのTPを基準のパケットとして採用している。
- ・ ISDBでもパケット長をTPと同様の長さに変更し、整合を取った。

長所

- ・ 伝送エラーの影響は最悪の場合でもPES単位でリセットされる。エラーの少ない場合はスライス単位で復帰。
- ・ 固定長化を行わないProgram Streamより、エラー後の復帰動作が確実である。

短所

- ・ 固定長化のためのスタッフィングにより、Program Streamより伝送効率が落ちる。

2. コンシールメントベクタ

- ・ フレーム内符号化モードの際にも動きベクタを付加して伝送するもの。

長所

- ・ 符号化データ中に伝送エラーがあった場合には、復号側でコンシールメントベクタを利用して動き補償をした補間画像を作成し、伝送エラーによる画質劣化を抑えることができる。

短所

- ・ 伝送すべきデータが増加する。

- ・ MPEGの規定にはないが、MPEGシステムとしては、以下のようなエラー対策の適用が考えられる。

3. 周期的イントラ符号化

- ・ MPEGでは時間軸方向圧縮を行なっているため、エラーの影響も時間方向に伝播する。この影響をある時間内に抑えるために、画面上のどの位置もそれぞれ周期的にイントラモード

で符号化を行なうことが必要。

4. 補間処理

- ・エラーが入り復号できないデータがある場合は、該当ブロックを前画面の同位置の画像データで置き換えたり、上下のブロックの動きベクタを利用して動き補償した前画面の画像データで置き換えるなどの補間処理により、受信画像の画質向上を図る必要がある。

5. 階層符号化

- ・階層符号化を行い、低域情報や低解像度情報のみ誤り対策を強化したり、品質の良い（セル廃棄等の少ない）伝送路で伝送するなどの方法で、エラーの影響が高域情報や高解像度情報に偏るようにして、エラー時でも低解像度の画像は再生できるようにする。

1 4. 蓄積系、コンピュータ応用等

1 4. 1 MPEGのビデオテープレコーダ記録

家庭用の蓄積装置に対するユーザー要求は、高画質/長時間録音、ならびに特殊再生機能である。MPEGにより符号化された映像データは、単独で復号できるI成分の他に時間軸方向に圧縮されたB成分やP成分から構成されている。このために、サーチ機能などの特殊再生を実現するには、GOP構成/符号化後のビットレートの他に、I成分を記録メディアへ如何に最適配置するかが不可欠になる。

本章では、特殊再生機能の実現を考慮した符号化データの蓄積方式に関して、その技術的特徴を従来のアナログ信号を対象とするVCRなどと比較しながら概観する。

(1) 従来のアナログ記録方式 (図14-1 (a))

入力映像信号の1フィールド分はFM変調した後に、ビデオテープ上の1トラックに記録する。1トラックにそれぞれ一画面を構成できる映像信号を記録しているために、早送り再生時にヘッドが各トラックを斜めに走査しても各トラックから検出できる信号をそのままFM変調すれば、多少のノイズバーが発生するがサーチの目的である記録内容の確認が可能である。

(2) 家庭用デジタルVCR^{[8][9][10]} (図14-1 (b))

特殊再生や1フレーム精度での編集を実現するためにフレーム内圧縮を基本としている。本VCRでは、デジタルコンポーネント信号をマクロブロック化し、圧縮効率を高めるためにマクロブロック単位でシャフリングしてからフレーム内圧縮処理を施す。その後、サーチ時の再生画像を見やすくするために再度並べ換えて(デシャフリング)、10本のトラック(525/60信号)に記録する。各トラックに記録されている映像信号の模式図を図14-1 (b)に示す。本記録方式によりサーチ時の画像は各トラックから1画面を構成する局所部分が再生されることになるのでブロック境界による妨害が若干発生するが(1)同様にサーチの目的を達成できる。

(3) MPEG符号化データ用VCR

符号化データを記録する方式には、大別して、単純にビットストリーマーとして記録する方式(図14-1 (c))と特殊用データをヘッドがスキャンするテープ上の位置に記録する方式(図14-1 (d))の2通りが考えられる。前者は、VCR側の処理が簡易なので、回路規模を小さくできる。しかし、イントラスライスのみを取り出して復号するため、画面の更新が短冊状になり、しかも隣り合うブロックの映像が数フレーム離れているため非常に見づらい画像となる。I成分をスキャンするようにテープを間欠的に動かすことで、画質は向上するが、機構系への負担が大きくなる。一方、後者は、I成分から低周波成分を抜き出すなどして生成した特再用データをヘッドがスキャンするテープ上の位置に記録するため、解像度の低い画像ではあるが、なめらかな早送り再生が像が得られる。しかし、前者に比べて、回路規模および記録データ量が大きくなるという欠点がある。

このように、特殊再生時の画質と機構系への負担、回路規模、記録データ量などがトレードオフの関係にあり、種々の検討が行われている。

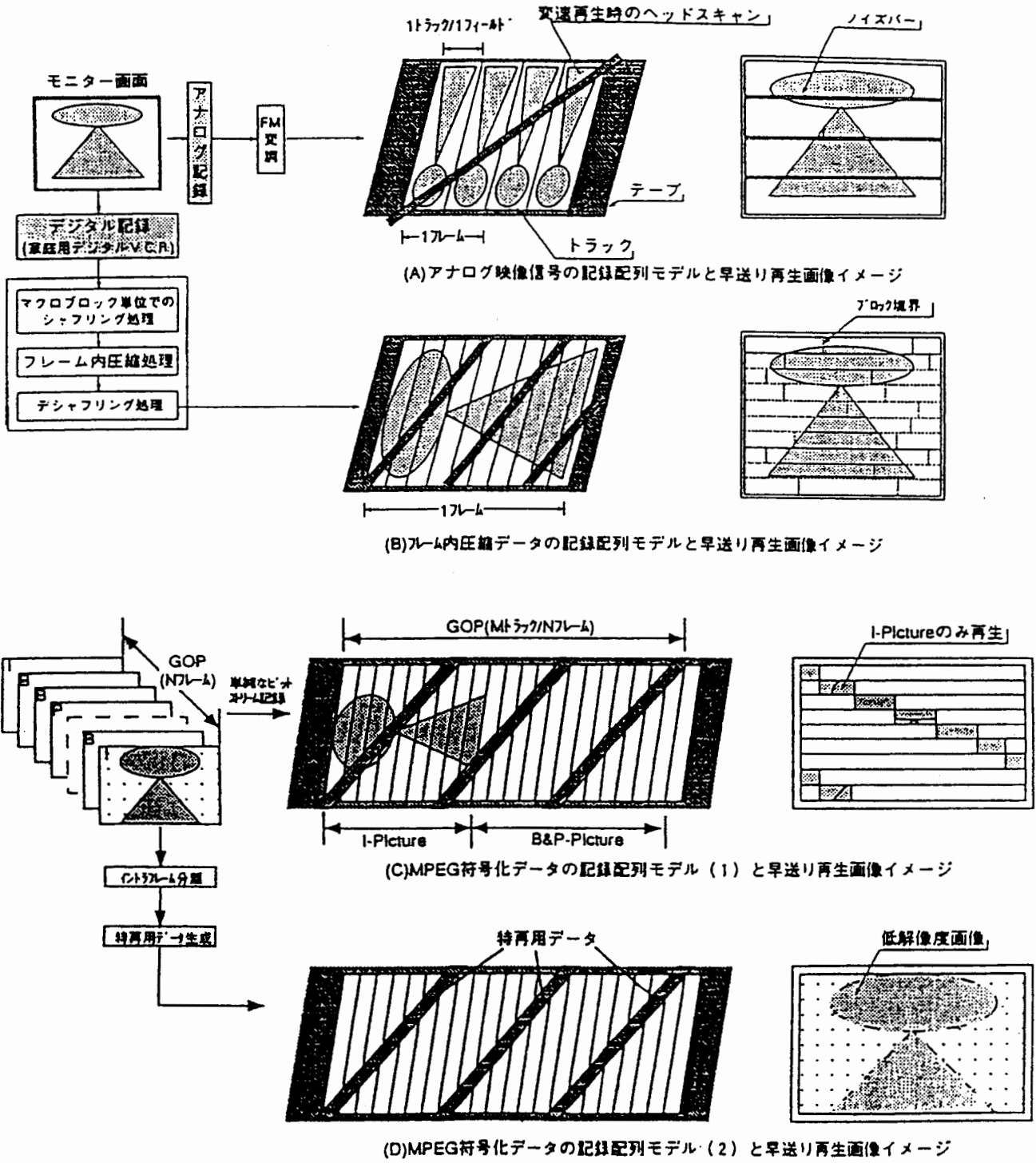


図14-1 MPEGのテープ記録

1 4 . 2 一般蓄積系・コンピュータ応用に対するMPEG2の考え方

MPEG2規格の基本的な標準化の思想は、

- ①インターオペラビリティ（相互運用性）
- ②スケーラビリティ（分解能拡張性）
- ③エクステンシビリティ（拡張性）

の3点にある。これにより、異なるメディアである通信・放送・蓄積系や異なるプラットフォーム（異機種コンピュータ）で情報交換が可能で、それぞれ必要とする表示系解像度で表示し、付加ビットの考え方で情報量を増加することが可能となった。

このようなメディア統合の考え方から、MPEG2では、蓄積系・コンピュータ応用として

- ①符号化ビットレートの柔軟性
- ②符号化・復号化遅延時間
- ③ランダム・アクセス
- ④ビットストリームの編集機能
- ⑤トリック・モード
- ⑥符号化・復号化の反復
- ⑦ATMへの接続
- ⑧スクエア・ピクセル

などの要求に対応している。

【符号化ビット・レートの柔軟性】

CBR (Constant Bit Rate) とVBR (Variable Bit Rate) の両方をサポートしている。用途により使い分けられる。

【符号化・復号化遅延時間】

テレビ電話などの遅延時間が問題になるアプリケーションでも、150msec以下を実現できる。たとえば、インターレースの入力信号について、Bピクチャを使わず、フィールド構造のみ使用するという低遅延モードが用意されている。

【ランダム・アクセス】

MPEG2では、GOP構造をうまく用いれば、蓄積系などのアプリケーションで好きなアドレスにアクセスできるように、データ構造は多くの時分割多重方式で用いられているパケット構造となっている。このため、用意されているシーケンス・ヘッダをリサーチすることにより、ランダム・アクセスが可能となる。(MPEG1の基本機能)

また、ランダム・アクセスをさらに容易にするため、アプリケーションの付加情報として、ディレクトリ情報や個別のストリームごとの種別情報が用意されている。(MPEGシステムの基本性能)

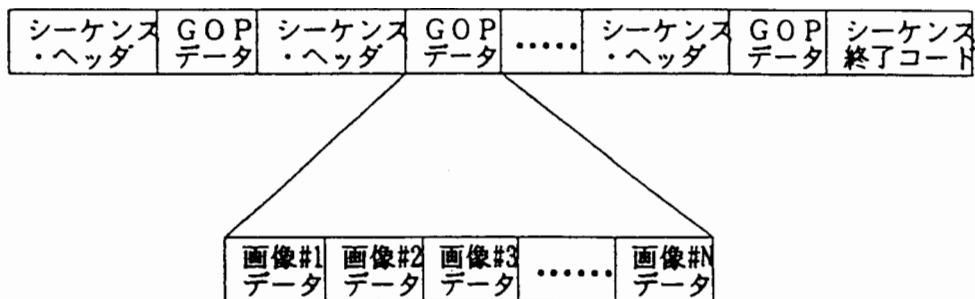


図14-2 GOP構造とシーケンス・ヘッダ

【ビットストリームの編集機能】

ランダム・アクセスと同じでGOP単位で編集することが可能。

【トリック・モード】

早送り、巻き戻し、スロー再生などのアナログVTRで実現できている機能を強力に実現するために、MPEG2では、スライスをイントラ（ピクチャ内符号化）・マクロブロックだけにして、トリック・モード時の再生画像とできるようにフラグを追加している。

【符号化・復号の反復による劣化】

編集やプログラム分配のために、符号化・復号を繰り返し行っても画像品質があまり劣化しないようにツールを提供している。

【ATMとの接続】

ATM (Asynchronous Transfer Mode) への接続を容易にするため、パケット長を可変としている。ATMで用いられる53バイトのパケット長から、光ディスクの4096バイトのパケット長に対応できるように、MPEGでは上限を約 2^{16} (64Kバイト) までフレキシブルにでき、各パケットごとに可変長でも固定長でも可能となっている。

【スクエア・ピクセル】

シーケンス・ヘッダ中、シーケンス・ヘッダ開始コードのあとに、画面の水平サイズ、垂直サイズ、画素アスペクト比などがデータとして送信可能なように規定されている。画素アスペクト比は4ビットで指定され、VGA (Video Graphics Array) などを対象にして、アスペクト比1.0000を用いることができるようになっている。これにより、コンピュータ・グラフィクスなどで使われるスクエア・ピクセルに対応可能である。

15. その他

15.1 データパーティショニング

【概要】

- ・誤り耐性を持たせる目的で、ビットストリームを二つに分ける手法を、データ・パーティショニングという。二つのビットストリームは、復号する前に再結合する。
- ・データ・パーティショニングは、複数のチャンネルを設けることが可能な伝送路や蓄積メディアで、使用可能なツールである。
- ・たとえば、ヘッダや動きベクトル、DC係数など、重要なビットストリームは、誤り特性の良いチャンネルで伝送、または、蓄積することによって実現する。
- ・これによって、チャンネル誤りに対し、緩やかな画像劣化特性、いわゆる、グレースフル・デグラデーション特性を持たせることが可能となる。
- ・どちらのチャンネルも、データ・パーティショニングを意図しないデコーダでの復号は不可能となる。
- ・シンタックスは、次の通りである。

```
sequence_scalable_extension()  
.....  
scalable_mode  
.....----->  
  
->(slice)  
.....  
if(scalable_mode=="data partitioning")  
    priority_breakpoint -----> (7ビット整数)  
.....
```

表15-1参照

マクロブロックと、ブロックレベルにはシンタックスはない。デコード処理でブレイクポイントに出会ったときには、以下の処理を行う。

- ① 次のパーティションが現パーティションになる。
- ② 現PBP(Priority Break Point)をストアする。
- ③ PBPは新しいブレイクポイントにセットされる。

逆転の場合は逆の動作をする。

表15-1 PBPと優先クラス

PBP	優先クラス	優先クラスに含まれるデータ
65	0	PBPまでのシーケンス、GOP、ピクチャ、スライス層の全データ
66	1	MBスタッフィングで始まり、MBタイプで終わるMBデータ
67	2	最初の動きベクトルまでのデータ
68	3	残りの動きベクトル
0	4	CBPに始まり、DC係数または最初の非ゼロ係数までのMBデータ
1	5	スキャン順のDCまたは最初の非ゼロ係数に続く最初の係数
2	6	スキャン順のDCまたは最初の非ゼロ係数に続く最初の係数の次の係数
.	.	
j	j+4	スキャン順のDCまたは最初の非ゼロ係数に続く(j-1)番目の係数の次のj番目の係数

- ・ PBPは、現パーティションの優先クラスを表している。
- ・ PBP=1以降、非ゼロ係数が無い場合のパーティションには、EOBをおく。
- ・ 非分割の場合には、PBP=63とする。

15.2 ACリーク

【概要】

- ・MPEG2には採用されていない。
- ・AC成分のリフレッシュをIピクチャに依存せず、Pピクチャの予測時で且つAC成分のみ、0.125または0.0625程度をリークする。
- ・これにより、データ量のIピクチャ集中がなくなり、発生データ量が平均化される。
- ・ノイズが混入した場合の画像劣化を軽減できる。

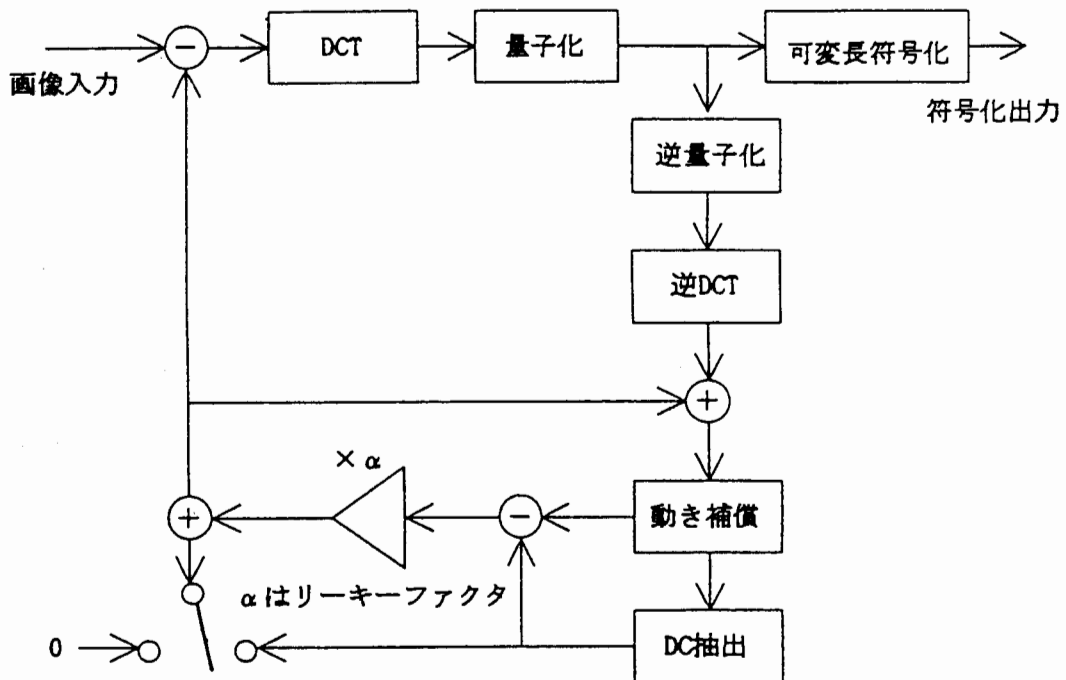


図15-2 ACリークの符号化装置

処理ステップ

- ①DC値の計算 (マクロブロックで6個)
- ②DC値を減算し、AC成分を求める。
- ③リーキー・ファクタ、0.875または0.9375をAC成分にかける。
- ④非リフレッシュ・ピクチャの場合は、AC値にDC成分を加算する。MC予測にはこの値を使う。マクロブロックタイプの決定は、DC成分の抽出以前におこなう。
- ⑤DCリフレッシュのために、周期的にDC値加算をやめる。DC成分は、イントラとして符号化され、DCT係数にはイントラVLCが適用される。

【特徴】

- ・SNRの改善はないか、少し劣化するが、周期的ACリフレッシュがないため、画質は向上する。DC値計算を必要とし、処理が複雑になる。

15.3 ローディレイモード

【概要】

テレビ電話・テレビ会議・監視等のリアルタイム通信システムでは、システム全体でのトータル遅延が少ない符号化方式が必要とされる。このような低遅延を目的とした符号化モードをローディレイモードという。

符号化／復号化における遅延の主な要因と低遅延実現のための手法を以下に示す。

(1) ピクチャの順番を入れ替える(ordering/reordering)ことによる遅延

低遅延化手法

- ・ Bピクチャを使わない。

特徴

- ・ 符号化効率が悪い。

(2) バッファ遅延

低遅延化手法

- ・ Iピクチャを使わずに、Iスライス(イントラスライス)を使う。

特徴

- ・ 動き補償予測の範囲に制限が必要である。(図15-3)
- ・ Iスライスと他のスライスにおける画質に差がないようにする必要がある。

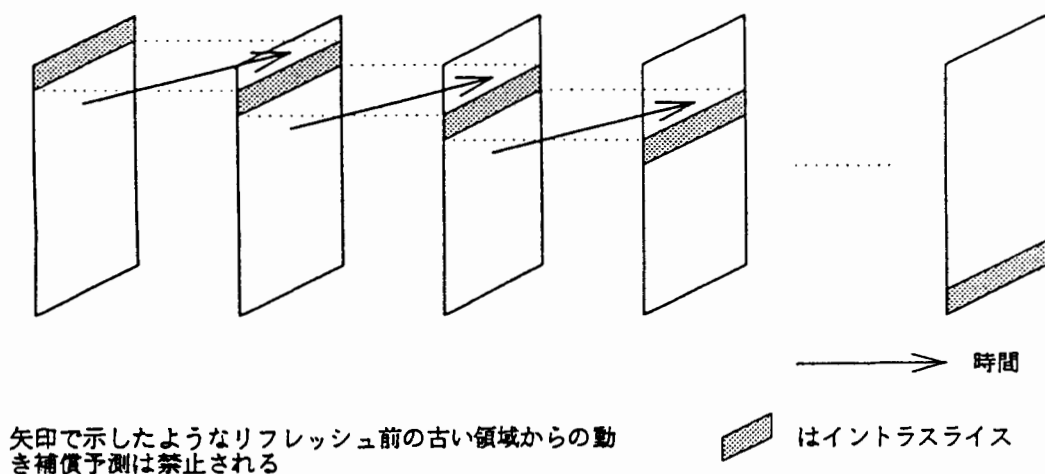


図15-3 イントラスライスを用いた場合の動き補償予測範囲の制限

規格では、Sequence extension 中の low_delay フラグがセットされているとき、そのシーケンスは以下のようなローディレイモードであることを示している。

- ・ Bピクチャを含まない。
- ・ VBVバッファのアンダーフローが起こりうる。(skipped picture が起こりうる。)

15.4 タイムスタンプ

MPEG2システム規格は、ビデオやオーディオ、更には他のデータを、1つあるいは複数のビットストリームに多重化する、あるいはその多重化されたビットストリームから各種データを分離する方式を定めた規格である。また、そのような各種データの同期再生の方式を定めた規格である。

タイムスタンプは、このMPEG2システム規格にて規定されたビットストリームに記述された情報の1つであり、ビットストリームに存在する各種データの同期をとるために用いられる。

タイムスタンプを用いた同期システムの基本概念図を図15-4に示す。

マルチプレクスされた符号化データは、デマルチプレクサにおいて、ビデオとオーディオの符号化データに分離されるとともに、タイムスタンプが抽出される。このタイムスタンプはビデオとオーディオの出力時刻を示しており、これに見合うように遅延バッファを制御することにより、両者の出力タイミングをとることができる。

ところで、MPEG2ビデオ規格では、GOP構造に依存して蓄積や伝送順序が変わるため、デコーダは内部に更にバッファを所有することになる。

そのため、実際のMPEG2システム規格では、タイムスタンプとして、プレゼンテーション・タイム・スタンプ(PTS)とデコーディング・タイム・スタンプ(DTS)の2つを定義している。

PTSは、デコーダにプレゼンテーション時刻を知らせるタイムスタンプであり、DTSは、デコーダにデコーディング時刻を知らせるタイムスタンプである。PTSとDTSは、90kHz単位で記述されている。

一方、デコーダでは、PTSとDTSの基準となるクロックを再生する必要がある。このようなデコーダのPLL(Phase Locked Loop)を行うために用いられるタイムスタンプとして、トランスポート・ストリームではPCR(Program Clock Reference)が、プログラム・ストリームではSCR(System Clock Reference)が定義されている。MPEG2システム規格では、PCRとSCRは、27MHz単位に拡張されている。

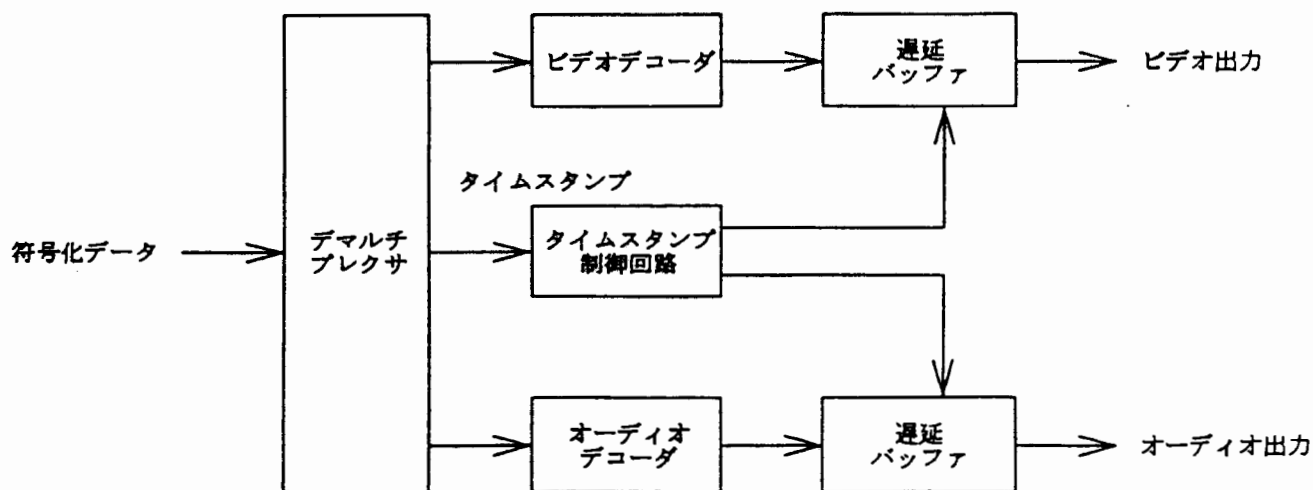


図15-4 タイムスタンプによる同期システムの基本概念

参考文献

- [1] 「データ圧縮とデジタル変調」日経エレクトロニクスブックス
- [2] 最前線レポート” デジタルテレビ” 日経ニューメディア 別冊
- [3]片山泰男：「MPEGの概要と標準化動向」 インターフェイス 1992年8月号
- [4]安田浩編：「マルチメディア符号化の国際標準」 丸善
- [5]渡辺裕：MPEG2/H. 262, TV誌、Vol. 48、No. 1、pp. 44-49 (1994)
- [6]米満監修：「MPEG技術」, (株)トリケップス, (1993)
- [7]渡辺 裕：「MPEG-2の標準化動向」情処研報Vol. 94, No. 53(94-AVM-5)、June 1994
- [8]井手 他：「家庭用デジタルVTR まず現行テレビ信号用規格を固める」
日経エレクトロニクス、1993. 10. 1 (No. 591)、pp. 137~150
- [9]三好 敏：「家庭用ハイビジョンVTR、二つの規格が固まる」
日経エレクトロニクス、1993. 10. 11 (No. 592)、pp. 115~121
- [10]高倉 他：「デジタルVTRの変速再生画質に関する検討」TV学会誌、
Vol. 47, No. 2、pp. 243~247
- [11]坂崎 他：「米国ATV対応デジタルVTRシステム検討」TV学会誌、
VR93-55、pp. 7~13
- [12]竹内 他：「高圧縮型デジタルVTRの高速再生方式の検討」TV学会誌、
VR93-57、pp. 21~26
- [13]ISO/IEC DIS 13818-2: Information Technology - Generic Coding of moving pictures and
associated audio information - Part 2: Video
- [14]片山泰男：「MPEG標準化会議 シドニー/ニューヨーク会合報告」
インターフェイス 1993年10月号

付録

ビデオ・シーケンス層

video_sequence() {	No. of bits	Mnemonic
next_start_code()		
sequence_header()		
if(nextbits()==extension_start_code) {		
sequence_extension()		
do {		
extension_and_user_data(0)		
do {		
if(nextbits()==group_start_code) {		
group_of_pictures_header()		
extension_and_user_data(1)		
}		
picture_header()		
picture_coding_extension()		
extension_and_user_data(2)		
picture_data()		
} while((nextbits()==picture_start_code)		
(nextbits()==group_start_code))		
if(nextbits()==sequence_end_code) {		
sequence_header()		
sequence_extension()		
}		
} while(nextbits()!=sequence_end_code)		
} else {		
/* ISO/IEC 11172-2 */		
}		
sequence_end_code		
}		

sequence_header() {	No. of bits	Mnemonic
sequence_header_code	32	bslbf
horizontal_size_code	12	unimsbf
vertical_size_code	12	unimsbf
aspect_ratio_information	4	unimsbf
frame_rate_code	4	unimsbf
bit_rate_value	18	unimsbf
marker_bit	1	"1"
vbv_buffer_size_value	10	unimsbf
constrained_parameter_matrix	1	
load_intra_quantiser_matrix	1	
if(load_intra_quantiser_matrix)		
intra_quantiser_matrix[64]	8*64	unimsbf
load_non_intra_quantiser_matrix	1	
if(load_non_intra_quantiser_matrix)		
non_intra_quantiser_matrix[64]	8*64	unimsbf
next_start_code()		
if(next_start_code() != extension_start_code) {		
if(next_start_code() == user_data_start_code)		
user_data()		
}		
}		

sequence_extension() {	No. of bits	
extension_start_code	32	bslbf
extension_start_code_identifier	4	unimsbf
profile_and_level_indication	8	unimsbf
progressive_sequence	1	unimsbf
chroma_format	2	unimsbf
horizontal_size_extension	2	unimsbf
vertical_size_extension	2	unimsbf
bit_rate_extension	12	unimsbf
marker_bit	1	"1"
vbv_buffer_size_extension	8	unimsbf
low_delay	1	unimsbf
frame_rate_extension_n	2	unimsbf
frame_rate_extension_d	5	unimsbf
next_start_code()		
}		

extension_and_user_data(i) {	No. of bits	
while((nextbits()==extension_start_code)	32	bslbf
(nextbits()==user_data_start_code)) {	4	unimsbf
if(nextbits()==extension_start_code)	8	unimsbf
extension_data(i)	8	unimsbf
if(nextbits()==user_data_start_code)	8	unimsbf
user_data()	1	unimsbf
}	2	unimsbf
}	5	unimsbf

GOP層

group_of_pictures_header()	No. of bits	Mnemonic
group_start_code	32	bslbf
time_code	25	
closed_gop	1	
bloken_link	1	
next_start_code()		
}		

ピクチャ層

picture_header() {	No. of bits	Mnemonic
picture_start_code	32	bslbf
temporal_reference	10	unimsbf
picture_coding_type	3	unimsbf
vbv_delay	16	unimsbf
if(picture_coding_type==2 picture_coding_type==3) {		
full_pel_forward_vector	1	
forward_f_code	3	unimsbf
}		
if(picture_coding_type==3) {		
full_pel_backward_vector	1	
backward_f_code	3	unimsbf
}		
while(nextbits()=='1') {		
extra_bit_picture	1	"1"
extra_information_picture	8	
}		
extra_bit_picture	1	"0"
next_start_code()		
}		

picture_coding_extension() (No. of bits	
extension_start_code	32	bslbf
extension_start_code_identifier	4	unimsbf
forward_horizontal_f_code	4	unimsbf
forward_vertical_f_code	4	unimsbf
backward_horizontal_f_code	4	unimsbf
backward_vertical_f_code	4	unimsbf
intra_dc_precision	2	unimsbf
picture_structure	2	unimsbf
top_field_first	1	unimsbf
frame_pred_frame_dct	1	unimsbf
concealment_motion_vectors	1	unimsbf
q_scale_type	1	unimsbf
intra_vlc_format	1	unimsbf
alternate_scan	1	unimsbf
repeat_first_field	1	unimsbf
chroma_420_type	1	unimsbf
progressive_frame	1	unimsbf
composite_display_flag	1	unimsbf
if(composite_display_flag) {		
v_axis	1	unimsbf
field_sequence	3	unimsbf
sub_carrier	1	
burst_amplitude	7	unimsbf
sub_carrier_phase	8	unimsbf
}		
next_start_code()		
}		

picture_data() {	No. of bits	Mnemonic
do {		
slice()		
} while(nextbits()==slice_start_code)		
next_start_code()		
}		

スライス層

slice()	No. of bits	Mnemonic
slice_start_code	32	bslbf
if(vertical_size>2800)		
slice_vertical_position_extension	3	unimsbf
if(<sequence_scalable_extension() がビットストリーム中にある>)		
if(scalable_mode=="data partitioning")		
priority_breakpoint	7	unimsbf
quantiser_scale_code	5	unimsbf
if(nextbits()=="1") {		
marker_bit	1	"1"
intra_slice	1	unimsbf
reserved_bits	7	unimsbf
while(nextbits()=="1") {		
extra_bit_slice	1	"1"
extra_information_slicce	8	
}		
}		
extra_bit_slice	1	"0"
do {		
macroblock()		
} while(nextbits()=="000 0000 0000 0000 0000 0000")		
next_start_code()		
}		

マクロブロック層

	No. of bits	Mnemonic
<code>macroblock() {</code>		
<code>while(nextbits()=='0000 0001 000')</code>		
<code>macroblock_escape</code>	11	vlclbf
<code>macroblock_address_increment</code>	1-11	vlclbf
<code>macroblock_modes()</code>	...	
<code>if(macroblock_quant)</code>		
<code>quantiser_scale_code</code>	5	unimsbf
<code>if(macroblock_motion_forward </code>		
<code>(macroblock_intra && concealment_motion_vectors))</code>		
<code>motion_vectors(0)</code>
<code>if(macroblock_motion_backward)</code>		
<code>motion_vectors(1)</code>
<code>if(macroblock_intra && concealment_motion_vectors)</code>		
<code>marker_bit</code>	1	"1"
<code>if(macroblock_pattern)</code>		
<code>coded_block_pattern()</code>
<code>for(i=0;i<block_count;i++) {</code>		
<code>block(i)</code>
<code>}</code>		
<code>}</code>		

	No. of bits	Mnemonic
<code>macroblock_modes() {</code>		
<code>macroblock_type</code>	1-9	vlclbf
<code>if((spatial_temporal_weight_code_flag==1) </code>		
<code>(spatial_temporal_weight_code_table_index!="00")) {</code>		
<code>spatial_temporal_weight_code</code>	2	unimsbf
<code>}</code>		
<code>if(macroblock_motion_forward </code>		
<code>macroblock_motion_backward) {</code>		
<code>if(picture_structure=='frame') {</code>		
<code>if(frame_pred_frame_dct==0)</code>		
<code>frame_motion_type</code>	2	unimsbf
<code>} else {</code>		

field_motion_type	2	unimsbf
)		
)		
if(decode_dct_type) {		
dct_type	1	unimsbf
}		
}		

ブロック層

block(i) {	No. of bits	Mnemonic
if(pattern_code[i]) {		
if(macroblock_intra)		
if(i<4) {		
dct_dc_size_luminance	2-9	vlclbf
if(dct_dc_size_luminance!=0)		
dct_dc_differential	1-11	unimsbl
} else {		
dct_dc_size_chrominance	2-10	vlclbf
if(dct_dc_size_chrominance!=0)		
dct_dc_differential	1-11	unimsbf
}		
} else {		
First DCT coefficient	...	
}		
while(nextbits() != End of block)		
Subsequent DCT coefficients	...	
End of block	...	
}		
}		

表中の略号は以下の意味である

bslbf Bit string, left bit first

unimsbf Unsigned integer most significant bit first

simsbf Signed integer most significant bit first

vlclbf Variable length code left bit first

略号一览

ATM	Asynchronous Transfer Mode
CBR	Constant Bit Rate
DCT	Discrete Cosine Transform
DSM	Digital Storage Media
DTS	Decoding Time Stamp
EOB	End Of Block
GOP	Group of Picture
IDCT	Inverse Discrete Cosine Transform
MAE	Mean Absolute Error
MSE	Mean Square Error
PBP	Priority Break Point
PCR	Program Clock Reference
PTS	Presentation Time Stamp
PES	Packetized Elementary Stream
SCR	System Clock Reference
TP	Transport Packet
VBR	Variable Bit Rate
VBV	Video Buffer Verifier
VLC	Variable Length Code

索引

- 1/2画素精度動き補償, 20
- 10ビットスケーラビリティ, 32
- 2次元ハフマン符号化, 21
- 3/2プルダウン, 29
- 4 : 2 : 0, 26
- 4 : 2 : 2, 26
- Activity, 23
- ACリーク, 49
- ATM, 32, 46
- CBR, 45
- DCT, 1, 7, 11, 31
- DCTとDPCMの比較, 7
- DCTとWaveletの比較, 8
- DCTの短所, 7
- DCTの長所, 7
- DSM, 1
- EOB, 21
- GOP, 1, 3, 56
- IDCT, 13
- IDCTミスマッチ, 13, 14
- Level, 34
- MAE, 16
- MQANT, 14
- MSE, 16
- ordering, 1, 3, 50
- PBP, 47
- PES, 41
- Profile, 34
- reordering, 1, 3, 50
- SNRスケーラビリティ, 31
- start code, 38
- Stepサーチ, 16
- Transport Stream, 41
- VBR, 45
- VBV, 25, 50
- VCR, 43
- VLC, 21
- VLCテーブル, 22
- Wavelet, 8
- Zero Run Length, 13
- 後処理, 26
- インターレース, 8
- イントラスライス, 1, 43, 50
- 動き検出, 16
- 動きベクトルの検出方法, 16
- 動きベクトル符号化, 21
- エスケープコード, 21, 22
- エラー対策, 41
- 階層的動きベクトルサーチ, 17
- 階層符号化, 42
- 片方向予測, 19
- 可変長符号化, 21
- 基本レイヤ, 31
- 空間Activity値, 24
- 空間スケーラビリティ, 31
- グレースフル・デグラデーション, 31

グレースフル・デグラデーション, 31
クローズドGOP, 6
クロマスケーラビリティ, 32
コンシールメントベクタ, 41
コンパチビリティ, 40
サイマルキャスト, 40
シーケンス, 53
時間スケーラビリティ, 32
ジグザグスキャン, 1, 9
シャフリング, 43
周期的イントラ符号化, 41
修正ベクトル, 18
周波数スケーラビリティ, 32
シンタックス, 37
スキャン, 30
スクエア・ピクセル, 46
スケーラビリティ, 31, 35
スライス, 1, 58
セル廃棄, 32, 42
ゼロラン, 21
双方向予測, 19
タイム・コード, 1
タイムスタンプ, 51
ダイレクトフルサーチ, 17
ツリーサーチ, 16
データパーティショニング, 32, 47
適応ジグザグスキャン, 9
デッドゾーン, 13
デフォルト量子化マトリクス, 11
デュアルプライム予測, 18
テレスコピックサーチ, 17
パン, 30
ピクチャ, 1, 3, 56
ビットストリーム, 37, 47
ビデオシーケンス, 1
フィールド予測, 18
フルサーチ, 16
ブレイクポイント, 47
フレーム/フィールド適応DCTの概要, 8
フレーム予測, 18
ブローケン・リンク, 6
ブロック, 1
ブロックマッチング, 16
変換係数の符号化, 21
補強レイヤ, 31
前処理, 26
マクロブロック, 1, 8, 18, 23, 59
モスキートノイズ, 12
ランダム・アクセス, 45
リーキー・ファクタ, 49
リフレッシュ, 49
量子化, 11
量子化スケール, 15
量子化ステップ, 13
量子化マトリクス, 11
レートコントロール, 23
ローディレイモード, 50

本調査は、TV情報符号化委員会・高能率符号化作業班の委員が執筆し、Adhoc-4のメンバーがとりまとめた。以下に作業班とAdhoc-4の構成を示す。

主任	キャノン (株)	檜田 素一
委員	三洋電機 (株)	土金 孝一
委員	シャープ (株)	杉野 道幸
委員	ソニー (株)	北里 直久
委員	(株) 東芝	嵩 比呂志
委員	日本電気 (株)	勝部 良次
委員	日本電気(株)	藤田 聡
委員	パイオニア (株)	木村 智博
委員	(株) 日立製作所	鈴木 教洋
*委員	(株) 富士通ゼネラル	新妻 規夫
*委員	松下電器産業 (株)	角野 眞也
*委員	三菱電機 (株)	加瀬沢 正
委員	(株) RCA技術研究所	桑原 義明
*委員	池上通信機 (株)	斎藤 利也
委員	富士通 (株)	森松 映史
委員	キャノン (株)	鹿倉 明祐
委員	日立電子 (株)	臼井 修司
委員	日本放送協会	筒井 健夫
委員	日本放送協会	鹿喰 善明
*委員	(株) 東京放送	鈴木 常夫
委員	日本テレビ放送網 (株)	浦野 丈治
委員	(株) フジテレビジョン	中田 安優
事務局	放送技術開発協議会	工藤 隆二
事務局	放送技術開発協議会	松崎 志郎

*印は本調査をまとめたAdhoc-4のメンバーを示す。