

# 圧縮映像情報の物理データモデル

3W-5

友田政明 堀内優希 植村俊亮

奈良先端科学技術大学院大学 情報科学研究科

## 1. はじめに

データベース上で動画像情報を扱う場合、ふつう画像情報の圧縮が行なわれるが、このことを考慮した物理データモデルの研究はほとんど行なわれていない。本論文では、動画像データベースにおける格納データがMPEG1<sup>[1]</sup>圧縮映像である場合における物理データモデルの提案を行なう。

## 2. MPEG1の物理構造

### 2.1 システムストリーム

MPEG1-SystemStreamは、図1のような多層構造を持つ。

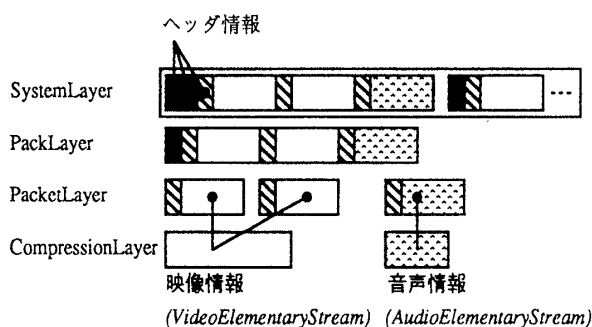


図1 SystemStreamの物理構造

### 2.2 映像情報

映像情報は、フレーム内符号化フレーム (I-picture)、フレーム間予測符号化フレーム (P-picture)、双方向予測符号化フレーム (B-picture) というそれぞれ符号化方式の異なる3種類のフレームから成るGOP(Group of pictures)で構成

されており、P,B-pictureは、それ自身では意味のある独立したフレームを復元することができない。

### 2.3 音声情報

音声情報は、対応する映像情報に同期して再生できるように、タイムスタンプ方式によって多重化されている。デコーダ側では、そのタイムスタンプ情報を基に映像情報との同期をとる。

## 3. MPEGデータモデル

MPEG1データベースにアクセスする場合、“物理的”な2次情報と“意味的”な2次情報の両者が必要になる。

### 3.1 オブジェクトデータモデル

MPEG1独特な物理構造に着目し、図2のようなオブジェクトデータモデルを提案する。

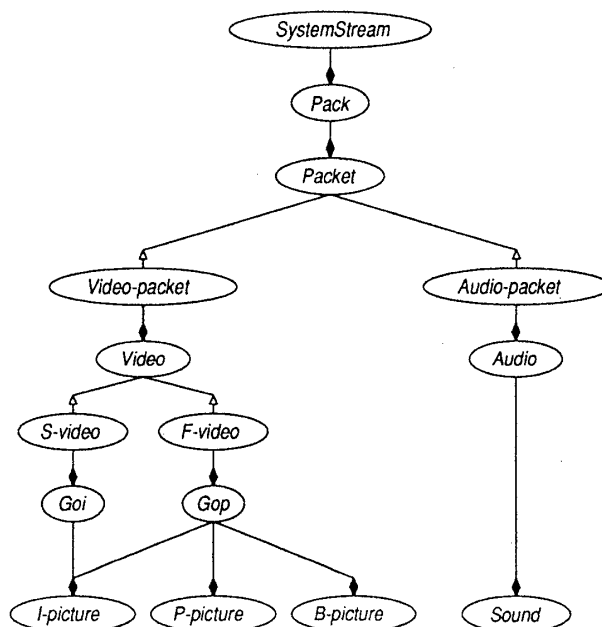


図2 物理データモデル

本データモデルの特徴を以下に示す。

- (1) *I,P,B-picture* それぞれ 1 フレームを一つのオブジェクトとする。
- (2) *I,P,B-picture* の順序集合を *Gop* オブジェクトとする。
- (3) *I-picture* の順序集合を *Goi* (*Group of I-pictures*) オブジェクトとする。
- (4) 同一再生時間区間に対応する *Video* と *Audio* を関連付ける。
- (5) 複数の *I,P,B-picture* (シーン) と一つの *Sound* を関連付ける。

### 3.2 MPEG オブジェクトのクラス定義

図 2 に基づいたクラス定義の例を以下に示す。オブジェクトの時間順序関連を指定するために順序集合 (*Ordered\_set\_of*) 属性を採用する。

- (1) *SystemStream* は *Pack* の順序集合で構成される。

```
CLASS SystemStream:{
  Ordered_set_of:
    Pack
}
```

- (2) *Video* と *Audio* の同期再生のためにタイムスタンプ属性を定義する。

```
CLASS Packet:{
  Is_a:
    Video-packet, Audio-packet
  PhysicalAttribute:
    time-stamp: set(Video-packet),
               set(Audio-packet)
}
```

- (3) *F-video* は *Gop* の順序集合で構成され、再生メソッドを持つ。

```
CLASS F-video:{
  Ordered_set_of:
    Gop
  PhysicalAttribute:
    component: list(Gop)
    start:     {I,P,B}-picture
    total:     INTEGER
    playback-time: INTEGER
  Method:
    PLAY
    FAST-FORWARD
    REWIND
    PAUSE
    JUMP
}
```

- (4) 編集 (*Gop* の順序変動) により孤立 *Gop* が発生するため、識別属性を定義する。

```
CLASS Gop:{
  Ordered_set_of:
    I-picture, P-picture, B-picture
```

```
PhysicalAttribute:
  component: list({I,P,B}-picture)
  flag:     {linked,closed,broken}
  previous: Gop
  next:     Gop
```

```
Method:
  PLAY
}
```

- (5) *Goi* は *I-picture* の順序集合で構成され、*Gop* と同様、再生メソッドを持つ。

```
CLASS Goi:{
  Ordered_set_of:
    I-picture
  PhysicalAttribute:
    component: list(I-picture)
    previous:  Goi
    next:      Goi
  Method:
    PLAY
}
```

- (6) *I-picture* は自身のみでフレームを復元できるため、再生メソッドを持つ。

```
CLASS I-picture:{
  PhysicalAttribute:
    previous: I-picture
    next:     I-picture
  Method:
    PLAY
}
```

- (7) *P,B-picture* は自身のみでフレームを復元不可能であるため、自身の属する *Gop* による疑似再生メソッドを持つ。

```
CLASS P-picture, B-picture:{
  PhysicalAttribute:
    Belong-to: Gop
  Method:
    PLAY by Belong_to
}
```

## 4. まとめと今後の課題

データベースで MPEG1 圧縮映像を管理するための物理データモデルの提案を行なった。今回は物理モデルに限ったが、これに意味的な 2 次情報を与えれば、さらに高水準のデータベース操作が可能となる。今後の課題としては、編集時における孤立 GOP の処理、OODB への実装等について検討を行ない、MPEG2 へと展開する。

### 参考文献

- [1] ISO/IEC 11172-1: 1993(E) Information technology - Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s - Part 1: Systems