

(19) 日本国特許庁(JP)

(12) 公表特許公報(A)

(11) 特許出願公表番号

特表2013-532441
(P2013-532441A)

(43) 公表日 平成25年8月15日(2013.8.15)

(51) Int.Cl.

HO4N 21/84 (2011.01)
HO4N 7/26 (2006.01)

F 1

HO4N 21/84
HO4N 7/13

テーマコード(参考)

5C159
5C164

審査請求 未請求 予備審査請求 未請求 (全 22 頁)

(21) 出願番号 特願2013-515413 (P2013-515413)
 (86) (22) 出願日 平成23年6月13日 (2011.6.13)
 (85) 翻訳文提出日 平成25年1月29日 (2013.1.29)
 (86) 國際出願番号 PCT/US2011/040168
 (87) 國際公開番号 WO2011/159605
 (87) 國際公開日 平成23年12月22日 (2011.12.22)
 (31) 優先権主張番号 61/354,422
 (32) 優先日 平成22年6月14日 (2010.6.14)
 (33) 優先権主張国 米国(US)
 (31) 優先権主張番号 61/354,424
 (32) 優先日 平成22年6月14日 (2010.6.14)
 (33) 優先権主張国 米国(US)

(71) 出願人 501263810
 トムソン ライセンシング
 Thomson Licensing
 フランス国, 92130 イッシー レ
 ムーリノー, ル ジヤンヌ ダルク,
 1-5
 1-5, rue Jeanne d' A
 r c, 92130 ISSY LES
 MOULINEAUX, France
 (74) 代理人 100115864
 弁理士 木越 力
 (74) 代理人 100121175
 弁理士 石井 たかし
 (74) 代理人 100134094
 弁理士 倉持 誠

最終頁に続く

(54) 【発明の名称】符号化マルチコンポーネント・ビデオをカプセル化する方法および装置

(57) 【要約】

複数のレイヤを含むメディア・エンティティを複数のコンポーネント・ファイルにカプセル化する(レイヤごとに1つのコンポーネント・ファイルにカプセル化する)方法および装置について、それに対応するコンポーネント・ファイルを読み取る方法および装置とともに説明する。ISO BMFF の新たなボックス、および SV C / MVC ファイル・フォーマットの抽出器データ構造の拡張を、提案する。この新たなボックスにより、参照したコンポーネント・ファイルに対するアクセスを、現在のコンポーネント・ファイルの処理と並行して行うことが可能になる。本発明の抽出器の拡張により、異なるコンポーネント・ファイルにまたがる NAL ユニットの参照が可能になる。本発明は、メディア・ファイルの適応型 HTTPストリーミングを可能にする。

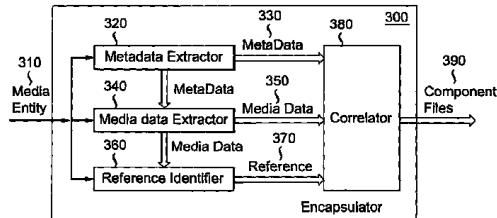


FIG. 3

【特許請求の範囲】**【請求項 1】**

複数のレイヤを含むメディア・エンティティからコンポーネント・ファイルを作成する方法であって、

前記メディア・エンティティから各レイヤのメタデータを抽出するステップと、

前記メディア・エンティティから、前記メディア・エンティティの各レイヤの前記抽出したメタデータに対応するメディア・データを抽出するステップと、

前記抽出したメディア・データを前記抽出したメタデータと関連付けて、前記抽出したメタデータおよび前記抽出したメディア・データを含むコンポーネント・ファイルを前記各レイヤに作成することを可能にするステップと、を含む、前記方法。

10

【請求項 2】

前記コンポーネント・ファイルが、動画ボックス、動画フラグメント、セグメントおよびファイルのうちの少なくとも1つである、請求項1に記載の方法。

【請求項 3】

前記各レイヤに、前記メディア・エンティティから、前記各レイヤの前記抽出したメディア・データに関係する追加メディア・データを抽出するステップと、

対応するコンポーネント・ファイルの作成のため、各レイヤの前記抽出したメディア・データと前記追加メディア・データとを関連付けるステップと、をさらに含む、請求項1に記載の方法。

【請求項 4】

各レイヤの前記抽出したメディア・データに関係する追加メディア・データの参照を識別するステップと、

対応するコンポーネント・ファイルの作成のため、前記参照を、各レイヤの前記抽出したメタデータおよび前記抽出したメディア・データと関連付けるステップと、をさらに含む、請求項1に記載の方法。

20

【請求項 5】

前記メディア・データおよび前記追加メディア・データが、データ・サンプルを含む、請求項4に記載の方法。

【請求項 6】

データ・サンプルが、ネットワーク抽象化レイヤ・ユニットを含む、請求項5に記載の方法。

30

【請求項 7】

前記参照が、前記追加メディア・データ中の前記ネットワーク抽象化レイヤ・ユニットの統一資源ロケータおよび統一資源名のうちの少なくとも1つを含む、請求項6に記載の方法。

【請求項 8】

前記参照を各レイヤの前記抽出したメタデータに埋め込むステップと、

前記参照の指標を前記抽出したメディア・データに付加するステップと、をさらに含む、請求項4に記載の方法。

【請求項 9】

前記参照が、各レイヤの前記コンポーネント・ファイルの冒頭に配置される、請求項8に記載の方法。

40

【請求項 10】

前記参照が、メディア参照ボックスに記入され、前記指標が、抽出器に記入される、請求項8に記載の方法。

【請求項 11】

複数のレイヤを含むメディア・エンティティからコンポーネント・ファイルを作成するファイル・カプセル化装置であって、

前記メディア・エンティティから各レイヤのメタデータを抽出し、前記メディア・エンティティから前記メディア・エンティティの各レイヤの前記抽出したメタデータに対応す

50

るメディア・データを抽出する抽出器と、

前記抽出したメディア・データを前記抽出したメタデータと関連付けて、前記抽出したメタデータおよび前記抽出したメディア・データを含むコンポーネント・ファイルを前記各レイヤに作成することを可能にする相関器とを含む、前記ファイル・カプセル化装置。

【請求項 1 2】

前記コンポーネント・ファイルが、動画ボックス、動画フラグメント、セグメントおよびファイルのうちの少なくとも 1 つである、請求項 1 1 に記載のファイル・カプセル化装置。

【請求項 1 3】

前記抽出器が、さらに、前記各レイヤに、前記メディア・エンティティから、前記各レイヤの前記抽出したメディア・データに関係する追加メディア・データを抽出し、

前記相関器が、さらに、各レイヤの前記抽出したメディア・データと前記追加メディア・データとを関連付けて、対応するコンポーネント・ファイルを作成する、請求項 1 1 に記載のファイル・カプセル化装置。

【請求項 1 4】

前記メディア・エンティティから、各レイヤの前記抽出したメディア・データに関係する追加メディア・データの参照を識別する参照識別器をさらに含み、前記相関器によって、前記参照を、各レイヤの前記抽出したメタデータおよび前記抽出したメディア・データと関連付けて、対応するコンポーネント・ファイルを作成する、請求項 1 1 に記載のファイル・カプセル化装置。

【請求項 1 5】

前記メディア・データおよび前記追加メディア・データが、データ・サンプルを含む、請求項 1 4 に記載のファイル・カプセル化装置。

【請求項 1 6】

データ・サンプルが、ネットワーク抽象化レイヤ・ユニットを含む、請求項 1 5 に記載のファイル・カプセル化装置。

【請求項 1 7】

前記参照が、前記追加メディア・データ中の前記ネットワーク抽象化レイヤ・ユニットの統一資源ロケータおよび統一資源名のうちの少なくとも 1 つを含む、請求項 1 6 に記載のファイル・カプセル化装置。

【請求項 1 8】

前記相関器が、さらに、前記参照を、各レイヤの前記抽出したメタデータに埋め込み、前記参照の指標を、前記抽出したメディア・データに付加する、請求項 1 4 に記載のファイル・カプセル化装置。

【請求項 1 9】

前記相関器が、前記参照を、各レイヤの前記コンポーネント・ファイルの冒頭に配置する、請求項 1 8 に記載のファイル・カプセル化装置。

【請求項 2 0】

前記参照が、メディア参照ボックスに記入され、前記指標が、抽出器に記入される、請求項 1 9 に記載のファイル・カプセル化装置。

【請求項 2 1】

コンポーネント・ファイルを読み取る方法であって、

前記コンポーネント・ファイルを構文解析して、メタデータ、メディア・データおよび参照を取得するステップと、

前記参照によれば、前記コンポーネント・ファイルの前記メディア・データがその他のコンポーネント・ファイルのメディア・データに関係している場合に、前記参照を用いて前記その他のコンポーネント・ファイルから前記関係するメディア・データを取り出すステップと、を含む、方法。

【請求項 2 2】

前記コンポーネント・ファイルの前記メディア・データが、符号化依存性に従ってその

10

20

30

40

50

他のコンポーネント・ファイルのメディア・データに関係付けられる、請求項 2 1 に記載の方法。

【請求項 2 3】

前記メディア・データおよび前記関係するメディア・データが、データ・サンプルを含む、請求項 2 1 に記載の方法。

【請求項 2 4】

データ・サンプルが、ネットワーク抽象化レイヤ・ユニットを含む、請求項 2 3 に記載の方法。

【請求項 2 5】

前記メタデータを構文解析して前記参照を取得するステップをさらに含む、請求項 2 1 10 に記載の方法。

【請求項 2 6】

前記メディア・データを構文解析して、その中に埋め込まれた参照指標を取得するステップと、

前記参照指標に従って対応する参照を取得するステップと、をさらに含む、請求項 2 5 に記載の方法。

【請求項 2 7】

前記取り出すステップが、

前記参照に従って、前記その他のコンポーネント・ファイルを並列に取り出すステップを含む、請求項 2 5 に記載の方法。 20

【請求項 2 8】

前記取り出すステップが、

ローカルなファイル記憶装置を確認するステップと、

前記ローカルなファイル記憶装置が前記その他のコンポーネント・ファイルを含む場合に、前記ローカルな記憶装置から前記その他のコンポーネント・ファイルを取り出すステップと、をさらに含む、請求項 2 7 に記載の方法。

【請求項 2 9】

コンポーネント・ファイルを構文解析して、メタデータ、メディア・データおよび参照を取得するパーサと、

前記参照に従って、その他のコンポーネント・ファイルから前記メディア・データに関係するメディア・データを取り出すリトリーバと、 30

前記メタデータ、前記メディア・データ、および前記その他のコンポーネント・ファイルから取り出したメディア・データを処理するプロセッサと、を含む、ファイル読み取り装置。

【請求項 3 0】

前記コンポーネント・ファイルの前記メディア・データが、符号化依存性の観点でその他のコンポーネント・ファイルのメディア・データに関係付けられる、請求項 2 9 に記載のファイル読み取り装置。

【請求項 3 1】

前記メディア・データおよび前記関係するメディア・データが、データ・サンプルを含む、請求項 2 9 に記載のファイル読み取り装置。 40

【請求項 3 2】

データ・サンプルが、ネットワーク抽象化レイヤ・ユニットを含む、請求項 3 1 に記載のファイル読み取り装置。

【請求項 3 3】

前記プロセッサが、ビデオ・デコーダを含む、請求項 2 9 に記載のファイル読み取り装置。

【請求項 3 4】

前記パーサが、前記参照を取得する手段をさらに含む、請求項 2 9 に記載のファイル読み取り装置。 50

【請求項 3 5】

前記パーサが、さらに、前記メディア・データを構文解析して、その中に埋め込まれた参照指標を取得し、前記参照指標に従って対応する参照を取得する、請求項 3 4 に記載のファイル読み取り装置。

【請求項 3 6】

前記リトリーバが、さらに、前記取得した参照に従って、前記その他のコンポーネント・ファイルを並列に取り出す、請求項 3 4 に記載のファイル読み取り装置。

【請求項 3 7】

前記リトリーバが、さらに、ローカルなファイル記憶装置を確認し、前記ローカルなファイル記憶装置が前記その他のコンポーネント・ファイルを含む場合には、前記ローカルな記憶装置から前記その他のコンポーネント・ファイルを取り出す、請求項 3 6 に記載のファイル読み取り装置。

10

【発明の詳細な説明】**【技術分野】****【0 0 0 1】**

(関連出願の相互参照)

本特許出願は、2010年6月14日出願の「Extension to the Extractor data structure of SVC/MVC file formats」と題する米国仮特許出願第61/354,422号、および2010年6月14日出願の「Some extensions for ISO Base Media File Format for HTTP streaming」と題する米国仮特許出願第61/354,424号の優先権の利益を主張するものである。

20

【0 0 0 2】

本願は、本願と同時に出願された「Method and Apparatus for Encapsulating Coded Multi-component Video」と題する、同時係属の所有者が共通の米国特許出願第__/_号（代理人番号PU100140）に関連するものである。

【0 0 0 3】

本発明は、一般に、HTTPストリーミングに関する。さらに詳細には、本発明は、HTTPストリーミングのスケーラブル・ビデオ符号化(SVC)ストリームやマルチビュー符号化(MVC)ストリームなどの符号化マルチコンポーネント・ビデオ・ストリームのメディア・エンティティのカプセル化に関する。

30

【背景技術】**【0 0 0 4】**

HTTPストリーミングの分野では、サーバ側で、MP4ファイルなどのBMFFに準拠したファイルとして、符号化ビデオをカプセル化して記憶することが多い。さらに、適応型HTTPストリーミングを実現するために、通常は、ファイルは、複数の動画フラグメントに分割され、これらのフラグメントが、さらに、クライアントURL要求によってアドレス可能な複数のセグメントにグループ化される。実際には、これらのセグメントには、ビデオ・コンテンツの様々な符号化表現が記憶され、クライアントが、所望の表現を動的に選択して、セッション中にダウンロードして再生するようになっている。

40

【0 0 0 5】

SVCやMVCのビットストリームなど、符号化された階層型ビデオは、ビットストリームの様々なサブセットを復号することによって、時間的/空間的解像度、画質、ビューなどに関して様々な動作点すなわち表現を可能にすることにより、このようなビットトレーナー適応の自然なサポートを実現する。しかし、MP4ファイル・フォーマットなど、既存のISO Base Media File Format(BMFF)規格は、各レイヤまたは表現への個別アクセスに対応していないので、HTTPストリーミング分野に適用することができない。図1に示すように、MP4ファイル・フォーマットでは、1つの

50

メディア・ファイルの全てのレイヤまたは表現のメタデータは、m o o v 動画ボックスに記憶され、一方、全てのレイヤまたは表現のメディア・コンテンツ・データは、m d a t 動画ボックスに記憶される。H T T P ストリーミングでは、クライアントが1つのレイヤを要求したとき、全てのレイヤまたは表現が混ざった状態であり、所要のレイヤまたは表現がどこで見つかるのかクライアントには分からないので、ファイル全体を送信しなければならない。

【0006】

後述のように、適応型H T T P ストリーミングの分野では、ネットワーク抽象化レイヤ(N A L)ユニットなどのメディア・データ・サンプルを、動画フラグメントまたはコンポーネント・ファイルの境界をまたいで参照することができる望ましい。S V C / M V C の状況では、このような参照は、「抽出器」などの機構を用いることによって実現することができる。「抽出器」は、以下のB M F F のA V C ファイル・フォーマット拡張に対するS V C / M V C 補正に定義される、内部ファイル・データ構造である。I n f o r m a t i o n T e c h n o l o g y - c o d i n g o f a u d i o - v i s u a l o b j e c t s - P a r t 1 5 : A d v a n c e d V i d e o C o d i n g (A V C) F i l e f o r m a t , A m e n d m e n t 2 : F i l e f o r m a t s u p p o r t f o r S c a l a b l e V i d e o C o d i n g , 2 0 0 8 , 1 5 ~ 1 7 ページ。抽出器は、コピーを行わずに、参照によってその他のトラックからN A L ユニットを抽出することを可能にするように設計されている。ここで、トラックとは、I S O ベース・メディア・ファイル中の関連するサンプルの時限シーケンスである。メディア・データでは、トラックは、画像またはサンプリングした音声のシーケンスに対応する。抽出器のシンタックスを、以下に示す。

```
class aligned(8) Extractor () {
    NALUnitHeader( );
    unsigned int(8) track_ref_index;
    signed int(8) sample_offset;
    unsigned int ((lengthSizeMinusOne + 1) * 8)
        data_offset;
    unsigned int ((lengthSizeMinusOne + 1) * 8)
        data_length;
}
```

【0007】

抽出器データ構造のセマンティクスは、以下の通りである。

N A L U n i t H e a d e r : タイプ20のN A L ユニットのI S O / I E C 1 4 4 9 6 - 1 0 A n n e x G に指定されるN A L ユニット構造 :

n a l _ u n i t _ t y p e は、抽出器N A L ユニット・タイプ(タイプ31)に設定されるものとする。

f o r b i d d e n _ z e r o _ b i t 、 r e s e r v e d _ o n e _ b i t およびr e s e r v e d _ t h r e e _ 2 b i t s は、I S O / I E C 1 4 4 9 6 - 1 0 A n n e x G に指定されるように設定されるものとする。

その他のフィールド(n a l _ r e f _ i d c 、 i d r _ f l a g 、 p r i o r i t y _ i d 、 n o _ i n t e r _ l a y e r _ p r e d _ f l a g 、 d e p e n d e n c y _ i d 、 q u a l i t y _ i d 、 t e m p o r a l _ i d 、 u s e _ r e f _ b a s e _ p i c _ f l a g 、 d i s c a r d a b l e _ f l a g およびo u t p u t _ f l a g)は、B . 4 o f I n f o r m a t i o n t e c h n o l o g y - C o d i n g o f a u d i o - v i s u a l o b j e c t s - p a r t 1 5 : A d v a n c e d V i d e o C o d i n g (A V C) f i l e f o r m a t , A m e n d m e n t 2 : F i l e f o r m a t s u p p o r t f o r S c a l a b l e V i d e o C o d i n g , I S O / I E C 1 4 4 9 6 - 1 5 : 2 0 0 4 / A m d . 2 : 2 0 0 8 , 1 7 ページに指定されるように設定されるものとする。

10

20

30

40

50

【0008】

`track_ref_index` は、データを抽出するトラックを発見するために使用するタイプ「`scal`」のトラック参照の指標を指定する。データが抽出されるトラック中のサンプルは、メディア復号タイムライン中で、すなわち時間 / サンプル・テーブルのみを使用して、抽出器を含むサンプルと時間的に整列される、またはそれより先行する最も近い位置に位置し、`sample_offset` によって指定されるオフセットだけ調節される。第1のトラック参照は、指標値1を有する。値0は予約される。

【0009】

`sample_offset` は、情報源として使用すべきリンクされたトラック中のサンプルの相対指標を与える。サンプル0(ゼロ)は、抽出器を含むサンプルの復号時間と比較して、同じ復号時間またはそれより先行する最も近い復号時間有するサンプルである。サンプル1(イチ)は、次のサンプルであり、サンプル-1(マイナス1)は、前のサンプルである。以下同様に続く。

【0010】

`data_offset` : コピーする参照サンプル内の最初のバイトのオフセットである。サンプル中のデータの最初のバイトから抽出が開始される場合には、オフセットは、値0をとる。オフセットは、NALユニット長さフィールドの冒頭を参照するものとする。

【0011】

`data_length` : コピーするバイト数である。このフィールドが値0をとる場合には、参照した1つのNALユニットの全体がコピーされる(すなわち、コピーする長さは、データ・オフセットによって参照した長さフィールドからとり、`Aggregator`の場合には`additional_bytes`フィールドで増補する)。

【0012】

さらなる詳細は、`Information technology - Coding of audio-visual objects - part 15: Advanced Video Coding (AVC) file format, Amendment 2: File format support for Scalable Video Coding, ISO/IEC 14496-15:2004/Amd.2:2008` に見ることができる。

【0013】

現在のところ、抽出器は、参照によってNALユニットを他のトラックから抽出することはできるが、同じ動画ボックス / フラグメント内から抽出することしかできない。換言すれば、抽出器を使用しても、別のセグメントまたはファイルからNALユニットを抽出することはできない。この制約により、抽出器の使用は上記の使用事例に制限されている。

【0014】

クライアントがサーバから1つのメディア・コンテンツの1つまたは複数のコンテンツ・コンポーネントを既にダウンロードしており、別のコンテンツ・コンポーネントをダウンロードするプロセスに入っている場合、クライアントは、必要に応じて完全なコンポーネント・セットをダウンロードするその他の要求を行うことができるよう、以前にダウンロードしたコンテンツ・コンポーネントが、新たなコンテンツ・コンポーネントの依存性コンポーネントのセットの中に含まれるかどうかを知る必要がある。この使用事例でも、外部依存性コンテンツ・コンポーネントおよびその位置情報を信号通信する機構が必要である。

【0015】

B M F F では、提示中の含んでいるトラックから別のトラックへの参照を実現するために使用される、「`treff`」と呼ばれるボックス・タイプがある。このボックスを使用して、トラック間の依存性を記述することができるが、その依存性は、同じメディア・ファイル内のトラックに限定される。

10

20

30

40

50

【0016】

1つの手法は、何らかの帯域外機構を用いてこの情報を信号通信するものである。例えば、H T T Pストリーミングの分野では、サーバは、セッションの開始前に、クライアントにマニフェスト・ファイルを送信することができる。マニフェスト・ファイルは、要求されたメディア・コンテンツの各コンテンツ・コンポーネントの依存性および位置情報を含むファイルである。この場合、クライアントは、必要な全てのコンポーネント・ファイルを要求することができる。ただし、この帯域外手法は、マニフェスト・ファイルが利用できないローカルなファイル再生には適用することができない。

【0017】

上述の問題に対する従来の解決策は、当技術分野ではまだ十分に確立されていない。速度およびトランスポート効率を犠牲にすることなく、複数のレイヤを構文解析してカプセル化することができるようになれば、望ましいであろう。このような成果は、当技術分野では、これまでのところまだ達成されていない。

10

【0018】

本発明は、複数のレイヤを含むメディア・エンティティからコンポーネント・ファイルをカプセル化し、コンポーネント・ファイルを読み取る、方法および装置に関する。

【0019】

本発明の1つの態様によれば、複数のレイヤを含むメディア・エンティティからコンポーネント・ファイルをカプセル化して作成する方法が提供される。この方法では、メディア・エンティティから、各レイヤに、メタデータを抽出し、その抽出したメタデータに対応するメディア・データを抽出する。抽出したメディア・データとメタデータとを関連付けて、抽出したメタデータおよび抽出したメディア・データを含むコンポーネント・ファイルを各レイヤに作成することを可能にする。

20

【0020】

本発明の別の態様によれば、ファイル・カプセル化装置が提供される。このファイル・カプセル化装置は、メディア・エンティティから各レイヤにメタデータを抽出し、且つ抽出したメタデータに対応するメディア・データを抽出する抽出器と、抽出したメディア・データを抽出したメタデータと関連付けて、各レイヤにコンポーネント・ファイルを作成することを可能にする相関器とを含む。

30

【0021】

本発明の上記の特徴は、添付の図面を参照しながらその例示的な実施例について詳細に説明することにより、さらに明らかになるであろう。

【図面の簡単な説明】

【0022】

【図1】例示的なM P 4ファイル・フォーマットを示す図である。

【図2】メディア・エンティティをカプセル化する本発明の一実施例を示す図である。

【図3】複数のレイヤ/表現を含むメディア・エンティティからコンポーネント・ファイルをカプセル化または作成するために使用されるカプセル化装置の構造を示す図である。

【図4】依存関係に基づいて追加メディア・データをコンポーネント・ファイルと関連付ける一例を示す図である。

40

【図5】抽出器が存在する動画ボックス/フラグメントとは異なる動画ボックス/フラグメントから参照によってN A Lユニットを抽出する一例を示す図である。

【図6】本発明の新たな抽出器データ構造の1つを用いた、S V C / M V Cタイプ・ビデオ・ビットストリームから複数のコンポーネント・ファイルへの関連するカプセル化動作を示す図である。

【図7】コンポーネント・ファイルを読み取るために使用されるファイル読み取り装置の構造を示す図である。

【図8】本発明の一実施例を含むビデオ・デコーダの、カプセル化されたコンポーネント・ファイルを読み取るプロセスを示す図である。

【図9】別的好ましい新たな抽出器データ構造を用いた、S V C / M V Cタイプ・ビデオ

50

・ビットストリームから複数の動画フラグメントへのカプセル化動作を示す図である。

【図10】本発明の別の実施例を含むビデオ・デコーダの、カプセル化されたコンポーネント・ファイルを読み取るプロセスを示す図である。

【発明を実施するための形態】

【0023】

本発明では、メディア・ファイルもしくはメディア・ファイル・セットまたはストリーミング・メディアなどのメディア・エンティティを、クライアントURL要求によってアドレス可能な複数の動画コンポーネント・ファイルに分割またはカプセル化する。ここでは、コンポーネント・ファイルは、フラグメント、セグメント、ファイル、およびその他のそれらと等価な用語を表す広い意味で用いられる。

10

【0024】

本発明の一実施例では、複数の表現またはコンポーネントを含むメディア・エンティティを構文解析して、各表現 / コンポーネントのメタデータおよびメディア・データを抽出する。この表現 / コンポーネントの例としては、様々な時間的 / 空間的解像度を有するレイヤなどのレイヤ、SVCの画質レイヤなどのレイヤ、およびMVCのビューなどがある。以下では、レイヤも、表現 / コンポーネントを指すために使用され、これらの用語は、入れ替え可能である。メタデータは、例えば、各表現のメディア・エンティティに何が含まれているか、またそこに含まれるメディア・データをどのように使用するかを記述している。メディア・データは、例えばコンテンツの復号など、メディア・データの目的を実施するのに必要なメディア・データ・サンプル、または所要のデータ・サンプルを取得する方法に関する任意の必要な情報を含む。各表現またはレイヤの抽出されたメタデータおよびメディア・データは、関連付け / 相関付けされ、ユーザのアクセスに備えて一緒に記憶される。記憶動作は、ハード・ドライブまたはその他の記憶媒体上で物理的に行ってよいし、あるいは、メタデータおよびメディア・データが実際には記憶媒体上の異なる場所に位置しているときでも、その他のアプリケーションまたはモジュールとのインターフェースをとったときに、それらが一緒に記憶されているように見えるように、関係管理機構を介して仮想的に実行してもよい。図2は、この実施例の例を示す。図2では、メディア・エンティティは、ベース・レイヤ、エンハンスメント・レイヤ1、およびエンハンスメント・レイヤ2の3つのレイヤを含む。メディア・エンティティを構文解析して、3つのレイヤそれぞれのメタデータおよびメディア・データを抽出し、これらのデータを、関連付けられたメタデータと対応するメディア・データとを備えるコンポーネント・ファイルとして、別々に記憶する。

20

【0025】

図3は、SVC符号化ビデオなど、複数のレイヤを含むメディア・エンティティからコンポーネント・ファイルをカプセル化して作成するために使用される好ましいカプセル化装置300の構造を示す。入力メディア・エンティティ310は、メタデータ抽出器320およびメディア・データ抽出器340に渡される。メタデータ抽出器320は、各レイヤのメタデータ330を抽出する。メディア・データ抽出器340は、メタデータ330を取り込み、対応するメディア・データ350を抽出する。なお、別の実施例では、メタデータ抽出器320およびメディア・データ抽出器340は、1つの抽出器として実施されることに留意されたい。メタデータ330およびメディア・データ350の両データは、これら2つのタイプのデータを関連付けて出力コンポーネント・ファイル390を作成する相関器380に送られる。コンポーネント・ファイルは、各レイヤに1つ作成される。

30

【0026】

SVCまたはMVCのAVC拡張によって符号化されたビデオなど、階層型ビデオは、複数のメディア・コンポーネント（スケーラブルなレイヤまたはビュー）を含む。このような符号化ビットストリームは、ビットストリームの様々なサブセットを復号することによって、時間的 / 空間的解像度、画質、ビューなどに関して様々な動作点すなわち表現またはレイヤを提供することができる。さらに、ビットストリームのレイヤ間には、符号化

40

50

依存性がある、すなわち、1つのレイヤの復号が、他のレイヤに依存する場合がある。従って、このようなビットストリームの表現のうちの1つを要求する際には、カプセル化ビデオ・ファイルから1つまたは複数のコンポーネントまたはメディア・データを取り出して復号する必要がある場合がある。様々な表現の抽出プロセスを容易にするために、符号化階層型ビデオは、各レイヤが異なるセグメントまたはコンポーネント・ファイルに別々に記憶されるような方法でMP4ファイルにカプセル化されることが多い。この場合には、上述の復号依存性またはアプリケーションによるその他の依存性によって、NALユニットなど、ビットストリームの特定のメディア・データ・サンプルが、複数のセグメントまたはコンポーネント・ファイルによって必要とされる、またはそれらと関係付けられることを考慮する必要がある。

10

【0027】

本発明の別の実施例では、セグメントまたはコンポーネント・ファイルによって必要とされる追加メディア・データを抽出して、当該セグメントまたはコンポーネント・ファイルと関連付ける。図4は、この実施例の例を示す。この図では、SVCビットストリームは、3つの空間レイヤ、HD1080p、SDおよびQVGAを有する。3つの動画フラグメントまたはコンポーネント・ファイルは、これら3つの動作点に対応して形成され、それぞれが異なるURLによってアドレス可能である。各動画フラグメントまたはコンポーネント・ファイル内で、復号に必要な全てのメディア・データ・サンプル（この例ではNALユニット）がコピーされ、「mdat」ボックスに収納されるメディア・サンプルとして記憶される。そのため、クライアントが適切なURLを用いて特定の動作点または表現を要求したとき、サーバは、対応する動画フラグメントまたはコンポーネント・ファイルを取り出して、クライアントに転送することができる。この実施例では、図3のメディア・データ抽出器340は、さらに各レイヤに、入力メディア・エンティティ310から、各レイヤに抽出したメディア・データに関する追加メディア・データを抽出する。相関器380は、さらに、この各レイヤの追加の抽出メディア・データを関連付けて、対応するコンポーネント・ファイルを作成する。

20

【0028】

記憶スペースを節約するために、各コンポーネント・ファイル内の同じデータを実際に複製することなく、複数の動画フラグメントまたはコンポーネント・ファイルの境界をまたいで、NALユニットなどのメディア・データ・サンプルを参照することができる事が望ましい。しかし、ISO Base Media File Format (BMFF) およびその拡張では、現在のところこの機能に対応していない。この問題を解決するために、本発明のさらに別の実施例では、動画フラグメントまたはコンポーネント・ファイルのメディア・データと関係付けられる、またはこれによって必要とされる追加メディア・データについて、参照を識別して構築する。これらの追加メディア・データではなく、この参照を、そのメタデータおよびメディア・データとともにコンポーネント・ファイルと関連付ける。参照は、各レイヤの抽出メディア・データに埋め込むことができ、その後、各レイヤの抽出メタデータおよび抽出メディア・データを関連付けて、対応するコンポーネント・ファイルを作成することができる。

30

【0029】

この実施例では、カプセル化装置300の構造に参照識別器360が追加される。参照識別器360は、入力メディア・エンティティ310から、各レイヤの抽出メディア・データ350に関するそれらの追加メディア・データに対する参照370を識別する。次いで、相関器380により、例えば参照370を抽出メディア・データ350に埋め込むことによって、参照370を各レイヤの抽出メタデータ330および抽出メディア・データ350と関連付けて、対応するコンポーネント・ファイル390を作成する。

40

【0030】

上述のように、SVC/MVCの状況では、このような参照は、「抽出器」などの機構を用いることによって構築することができる。現在のところ、抽出器は、参照によってNALユニットを他のトラックから抽出することはできるが、同じ動画ボックス/フラグメ

50

ント内から抽出することしかできない。換言すれば、抽出器を使用しても、別のセグメントまたはファイルから NAL ユニットを抽出することはできない。この制約により、他のケースの抽出器の使用は制限されている。以下、抽出器データ構造の拡張を開示するが、この拡張は、上述した SVC / MVC タイプの階層型ビデオ・コンテンツから複数のコンポーネント・ファイルへの効率的なカプセル化をサポートすることを目的とするものである。

【0031】

この拡張は、抽出器が存在する動画ボックス / フラグメントまたはコンポーネント・ファイルとは異なる動画ボックス / フラグメントまたはコンポーネント・ファイルに存在する NAL ユニットを参照する追加機能を有する抽出器データ構造を提供するために追加されるものである。

10

【0032】

拡張された抽出器は、以下のように定義される。

【0033】

シンタックス：

```
aligned (8) class DataEntryUrlBox (bit (24) flags)
  extends FullBox ('url', version = 0, flags) {
  string location;
}

aligned (8) class DataEntryUrnBox (bit (24) flags)
  extends FullBox ('urn', version = 0, flags) {
  string name;
  string location;
}

class aligned (8) Extractor () {
  NALUnitHeader ( );
  DataEntryBox (entry_version, entry_flags) data_entry; // added extension
  unsigned int(8) track_ref_index;
  signed int(8) sample_offset;
  unsigned int ((lengthSizeMinusOne + 1) * 8) data_offset;
  unsigned int ((lengthSizeMinusOne + 1) * 8) data_length;
}
```

20

30

【0034】

セマンティクス：

data_entry は、統一資源ロケータ (URL) または統一資源名 (URN) エントリである。名称は URN であり、URN エントリで必要である。位置は URL であり、URL エントリで必要であるが、URN エントリでは任意選択であり、所与の名称を有する資源が見つかる位置を与える。それぞれは UTF-8 文字を用いたナル終了文字列である。自立フラグがセットされた場合には、URL フォームを使用し、文字列は存在せず、ボックスはエントリ・フラグ・フィールドで終了する。URL タイプは、ファイルを配信するサービスのものとする。相対 URL は許容されるが、抽出器が属するトラックを含む動画ボックス / フラグメントを含むファイルに関するものである。

40

【0035】

その他のフィールドは、前述の元の抽出器と同じセマンティクスを有する。

【0036】

拡張された抽出器では、参照によって、抽出器が存在する動画ボックス / フラグメントとは異なる動画ボックス / フラグメントから、NAL ユニットを抽出することができる。図 5 は、このような例を示し、SVC ビットストリームは図 4 と同じであるが、新たな拡

50

張された抽出器データ構造を用いている。この図から分かるように、SDの動画フラグメントは、QVGAの動画フラグメントのNALユニットを参照することができる。同様に、HD1080pの動画フラグメントは、抽出器を使用して、QVGAおよびSDの両方の動画フラグメントのNALユニットを参照することができる。図4と比較すると、これらの動画フラグメントをまたいで複製されるNALユニットは存在せず、従って、記憶スペースが節約される。

【0037】

図6は、本発明の新しい抽出器データ構造を用いた、SVC/MVCタイプのビデオ・ビットストリームから複数の動画フラグメントまたはコンポーネント・ファイルへの関連するカプセル化動作を示す。このプロセスは、ステップ601で開始される。ステップ610で、各NALユニットを1つ1つ読み込む。ステップ620で、ビットストリームの末尾に到達した場合には、プロセスは、690で停止し、そうでない場合には、プロセスは、次のステップ630に進む。判断ステップ630では、現在のNALユニットが復号に際して他のトラックのNALユニットに依存するかどうかを判定する。この判定の結果が、現在のNALユニットが復号に際して他のトラックのNALユニットに依存しない場合には、制御はステップ640に移り、現在のNALユニットを用いてサンプルが形成され、現在のトラック中に配置される。ステップ630の判定の結果、現在のNALユニットと他のトラックのNALユニットの間に依存性がある場合には、プロセスは、ステップ650に進む。判断ステップ650では、現在のNALユニットによって必要とされるNALユニットのトラックが同じ動画フラグメント内に存在するかどうかをさらに判定する。この判定の結果、トラックが同じ動画フラグメント内に存在する場合には、ステップ670を利用して、拡張された抽出器に記入(fill in)して、当該他のトラックのNALユニットを参照する。判定の結果、トラックが別の動画フラグメント内に存在する場合には、ステップ660で、この動画フラグメントのURLまたはURNを識別し、この識別したURLおよびURNを拡張された抽出器に記入するものとして、プロセスは、ステップ670に進む。この拡張された抽出器は、記入後に、ステップ680で現在のトラックに埋め込まれる。その後、ステップ610で、次のNALユニットのプロセスが開始される。

10

20

30

40

【0038】

別の実施例では、相関器380によって、参照370が抽出メタデータ330に埋め込まれ、参照370の指標が抽出メディア・データ350に付加される。相関器380は、さらに、各レイヤのメタデータとメディア・データを関連付けて、対応するコンポーネント・ファイル390を作成する。ISO Media Base File Formatの状況では、HTTPストリーミング情報ボックスと呼ばれるボックスが開示されている。このボックスは、ISOファイルのHTTPストリーミングを補助することができる情報を含む。HTTPストリーミング情報ボックスは、例えばファイルの冒頭など、できる限り早い段階でコンポーネント・ファイル中に配置されることが好ましい。このボックスは、サーバがクライアントに対するマニフェスト・ファイルを形成するときのソースとして機能することもできる。HTTPストリーミング情報ボックスに含まれる、メディア参照ボックスと呼ばれる別のタイプのボックスも、開示されている。このボックスは、外部依存ファイルに関する情報を含む。抽出器構造は、さらに、異なるコンポーネント・ファイルをまたいで複数のメディア・サンプルを参照することができるよう拡張される。メディア参照ボックスに含まれる情報は、抽出器が信号通信のオーバヘッドを節約するために利用することができる。

【0039】

提案したHTTPストリーミング情報ボックス、メディア参照ボックスおよびさらに改良された抽出器の詳細な定義は、以下の通りである。

【0040】

HTTPストリーミング情報ボックス

定義：

50

ボックス・タイプ : 'hsin'
 コンテナ : ファイル
 必須 : いいえ
 量 : 0 または 1

【 0 0 4 1 】

H T T P ストリーミング情報ボックスは、I S O メディア・ファイルのH T T P ストリーミング動作を補助する。H T T P ストリーミング情報ボックスは、考えられる他のタイプのボックスの中でも特に、以下に定義するメディア参照ボックスを含む、ファイルのH T T P ストリーミング配信に関する関連情報を含む。H T T P ストリーミング情報ボックスは、可用性を最大限に高めるために、できる限り早い段階でファイル中に配置することが好ましい。

10

【 0 0 4 2 】

シンタックス :

```
aligned(8) class HTTPStreamingInfoBox extends Box ('hsin') {  
}
```

【 0 0 4 3 】

メディア参照ボックス

定義 :

ボックス・タイプ : 'mref'
 コンテナ : 'hsin'
 必須 : いいえ
 量 : 0 または 1

20

【 0 0 4 4 】

メディア参照ボックスは、H T T P ストリーミング情報ボックスに含まれ、当該ボックス内に含まれる各トラックが依存する外部ファイルの位置を宣言するU R L の形態をしたデータ参照のテーブルを含む。このボックスを読み取ることにより、ファイル読み取り装置は、外部コンポーネント・ファイルなど、ファイル中のトラックの外部依存ファイル・ソースと、それらを取り出すための手段とを識別することができる。

【 0 0 4 5 】

シンタックス :

```
aligned(8) class DataEntryUrlBox ( bit(24) flags ) extends Box ('url') {  
    string location;
```

30

```
}
```

```
aligned(8) class MediaReferenceBox extends Box ('mref') {  
    unsigned int(16) entry_count;  
    for ( i = 1; i <= entry_count; i++ ) {  
        unsigned int(32) track_ID;  
        unsigned int(16) dependent_source_count;  
        for ( j = 1; j <= dependent_source_count; j++ ) {  
            DataEntryUrlBox data_entry;  
        }  
    }  
}
```

40

【 0 0 4 6 】

セマンティクス :

e n t r y _ c o u n t は、実際のエントリをカウントした整数である。
 t r a c k _ I D は、ボックスが適用されるファイル中のトラックを一意に識別する整数である。

d e p e n d e n t _ s o u r c e _ c o u n t は、t r a c k _ I D を有するファイル中のトラックが依存する外部メディア・ソースをカウントした整数である。

50

`data_entry` は、指定されたトラックが依存する 1 つの外部メディア・ソースを指す URL エントリである。それぞれは、UTF-8 文字を用いたナル終了文字列である。URL タイプは、ファイルを配信するサービスのものとする。相対 URL は許容されるが、このメディア参照ボックスを含むファイルに関するものである。

【0047】

上記で定義したメディア参照ボックスは、複数のレイヤを含むメディア・エンティティの HTTP ストリーミングをいくつかの方法で容易にするように設計される。

【0048】

第 1 に、メディア参照ボックスは、参照テーブルによってコンポーネント・ファイルの冒頭でコンポーネント・ファイル間の依存関係を明示的に信号通信することができる。従って、クライアントは、コンポーネント・ファイルのわずかな部分を一度ダウンロードした後は、その 1 つまたは複数のトラックの関連する外部コンポーネント・ファイルを全て知ることができ、必要に応じて、テーブルに含まれる参照により、対応する要求を行って、1 つまたは複数の完全なセットを再生のために取得することができる。

10

【0049】

第 2 に、このボックスのファイル内情報は、容易に抽出して、マニフェスト・ファイルに含めることができる。この情報をマニフェスト中に含めることで、クライアントが実際の HTTP ストリーミングより前に、関連サービス情報を発見し、例えば全ての関連コンポーネント・ファイルの要求や必要なバッファ資源の割当てなど対応するサービス初期化を実行するのを助けることができる。

20

【0050】

第 3 に、クライアントは、コンポーネント・ファイルとして既に配信されている別の表現を有する、表現の異なる何らかのマルチコンポーネント・メディア・コンテンツを要求するときに、ファイル中の対応するメディア参照ボックスを確認して、そのファイルが再利用可能な新たな表現の任意の依存性コンポーネントを含むかどうかを調べることができる。

【0051】

最後に、このボックスは、以下のように定義された拡張された抽出器構造の信号通信のオーバヘッドを低減するのに役立つ。

30

【0052】

抽出器

抽出器は、さらに、外部メディア・ファイルのトラックのデータを参照する機能を拡張することが提案されている。

【0053】

拡張シンタックス：

```
class aligned(8) Extractor ( ) {
    NALUnitHeader ( );
    unsigned int(16) media_reference_index;
    unsigned int(8) track_ref_index;
    signed int(8) sample_offset;
    unsigned int ((lengthSizeMinusOne + 1) * 8)
        data_offset;
    unsigned int ((lengthSizeMinusOne + 1) * 8)
        data_length;
}
```

40

【0054】

セマンティクス：

`media_reference_index`：抽出器を含むトラックと同じ関連する `track_ID` の値を有するメディア参照ボックスに含まれる参照テーブルのエントリの指標を指定する。`media_reference_index` が 0 に等しい場合には

50

、抽出器は、別のトラックのデータではあるが、抽出器と同じファイル内のデータを参照する。この場合には、このトラックと同じ track_ID の値を有するメディア参照ボックス内の参照テーブルは存在しないものとする。media_reference_index が、1 から、メディア参照ボックスのトラックと関連付けられた参照テーブルの dependent_source_count の値までの間である場合には、参照テーブルの media_reference_index によって参照される URL が外部ファイルを指し、この外部ファイルが、抽出器がデータを抽出するトラックを含む。

【0055】

他のフィールドのセマンティクスは、元の抽出器の定義と同じままである。

【0056】

このさらに拡張された抽出器構造では、抽出器を使用して、外部コンポーネント・ファイルに属するトラックのデータにリンクしてこれを抽出することができる。これは、SVC や MVC で符号化されたものなど、1 つの符号化されたマルチコンポーネント・メディア・コンテンツからのコンテンツ・コンポーネントを異なるコンポーネント・ファイルにカプセル化するときに、特に有用である。この拡張された抽出器では、ファイルの境界をまたいで抽出を行うことができる。これにより、異なるコンポーネント・ファイル中の同じデータを複製しなくても済む。

【0057】

図 9 は、開示した HTTP ストリーミング情報ボックスおよびメディア参照ボックスならびに上記のさらに拡張された抽出器データ構造を用いた、SVC / MVC タイプのビデオ・ビットストリームから複数の動画フラグメントまたはコンポーネント・ファイルへの関連するカプセル化動作を示す。このプロセスは、上述のボックスおよび抽出器のさらなる拡張によるいくつかの修正点を除けば、図 6 に示すプロセスと同様である。ステップ 660 で位置情報 URL / URN が識別された後で、ステップ 965 で、この位置情報を使用して、mref ボックス（メディア参照ボックス）中の参照テーブルに記入を行う。ステップ 970 で、さらに、参照テーブルの位置情報の指標を抽出器に記入する。次いで、抽出器を現在のトラックに埋め込む。ステップ 620 でビットストリームの末尾に到達している場合には、mref ボックスおよびそのコンテナ hsin ボックス（HTTP ストリーミング情報ボックス）を、コンポーネント・ファイルのメタデータに埋め込む。

【0058】

コンポーネント・ファイルを読み取るために、図 7 に示すファイル読み取り装置 700 を利用する。パーサ 710 が、最初に、コンポーネント・ファイルを構文解析して、メタデータおよびメディア・データを取得し、入手可能であれば参照を取得する。復号した参照から、メディア・データが復号依存性などによって他のコンポーネント・ファイルのメディア・データに関係していることが分かった場合には、リトリーバ（retriever）720 が、この関連するメディア・データを、参照に示されている他のコンポーネント・ファイルから取り出す。さらに、プロセッサ 730 は、コンポーネント・ファイルから取得したメタデータおよびメディア・データを処理し、入手可能であれば追加メディア・データも処理する。パーサ 710 による構文解析動作は、プロセッサ 730 のために用意されたメタデータおよびメディア・データ、ならびにリトリーバ 720 のために用意された参照を取得するために必要な様々な動作を含む。この構文解析動作は、必要に応じて、メタデータおよび / またはメディア・データをさらに構文解析することを含む。一実施例では、参照は、メディア・データに埋め込まれるので、メディア・データを構文解析することによって取得される。参照が入手可能である場合には、構文解析ステップは、参照のシンタックスを分析すること、および参照を復号することをさらに含む。プロセッサ 730 は、コンポーネント・ファイルがビデオ・コンテンツを含む場合には、ビデオ・デコーダを含むことができる。別の実施例では、パーサおよびリトリーバをプロセッサに組み込むこともできる。

【0059】

図 8 は、本発明を含むビデオ・デコーダの SVC / MVC タイプ・ビデオ・ビットスト

10

20

30

40

50

リームを読み取るプロセスを示す。ステップ 801 で、コンポーネント・ビデオ・ファイルにアクセスし、ステップ 805 で、当該コンポーネント・ビデオ・ファイルの各レイヤのメタデータおよびメディア・データを識別する。ステップ 810 で、識別したメタデータおよびメディア・データを構文解析し、ステップ 815 で、メディア・データの各 NAL ユニットを 1 つ 1 つ読み込む。現在の NAL ユニットについて、ステップ 820 で最初に判断を行い、ビットストリームの末尾に到達しているかどうかを判定し、その結果が「はい」である場合には、プロセスは、ステップ 825 で終了する。そうでない場合には、プロセスは、判断ステップ 830 に進み、現在の NAL ユニットが抽出器であるかどうかを判定する。現在の NAL ユニットが抽出器ではない場合には、それは現在の NAL ユニットが復号用データを含む通常の NAL ユニットであることを意味するので、ステップ 835 で、この NAL ユニットをデコーダに送信する。現在の NAL ユニットが抽出器である場合には、ステップ 840 で、現在の NAL ユニットが同じコンポーネント・ファイル外の NAL ユニットに依存するかどうかを判定する。必要とされる NAL ユニットが同じコンポーネント・ファイル内にある場合には、ステップ 845 で、その NAL ユニットを現在のファイルから取り出し、ステップ 835 で、デコーダに送信する。必要とされる NAL ユニットが別のコンポーネント・ファイルのものである場合には、ステップ 850 で、抽出器中の参照情報 Data_directory を用いてその NAL ユニットを突き止め、ステップ 855 で遠隔ファイルから取り出し、その後、ステップ 835 でデコーダに送信する。

10

【0060】

20

別の実施例では、パーサ 710 がメディア・データを構文解析することによって参照を識別して、埋め込まれた参照指標を得、これらの参照指標に従って対応する参照を取得する。ビデオ・デコーダの SVC / MVC タイプ・ビデオ・ビットストリームを読み取る対応するプロセスを、図 10 に示す。これは、図 8 のプロセスと同様である。好ましい実施例ではコンポーネント・ファイルの冒頭に参照が配置されるので、ステップ 810 のメタデータの構文解析により、その中に含まれる参照の分析をメディア・データの構文解析と並行して行うことが可能になる。参照を分析するときには、ステップ 1014 で、参照されたその他のコンポーネント・ファイルを識別する。これらのその他のコンポーネント・ファイルの取出しは、ステップ 1012 で、このプロセスの残りのステップと並行して開始される。ステップ 850 で、現在の NAL ユニットが依存するコンポーネント・ファイルの位置情報にアクセスした後で、このコンポーネント・ファイル入手することができるかどうか、メディア・バッファなどのローカル記憶装置を確認する。必要なコンポーネント・ファイルがローカルで入手できる場合には、そのローカル・コピーの NAL ユニットを取り出し、そうでない場合には、遠隔ファイルの NAL ユニットを取り出す。なお、このコンポーネント・ファイルのローカル・コピーは、ステップ 1012 で並行取出しによって取得してもよいし、このコンポーネント・ファイルの以前の要求から取得してもよいことに留意されたい。

30

【0061】

40

本明細書では、本発明の好ましい実施例について詳細に説明したが、本発明はこれらの実施例に限定されるわけではないこと、および当業者なら、添付の特許請求の範囲に定義する本発明の範囲を逸脱することなく、その他の修正形態および変形形態を実施することができることを理解されたい。

【図1】

ftyp:ファイル・タイプ・ボックス
moov:動画ボックス ベース・レイヤ、エンハンスメント・レイヤ1およびエンハンスメント・レイヤ2などのメタデータ
mdat:メディア・データ・ボックス ベース・レイヤ、エンハンスメント・レイヤ1およびエンハンスメント・レイヤ2などのメディア・データ

図1

【図2】

エンハンスメント・レイヤ1 moov:動画ボックス
エンハンスメント・レイヤ1のメタデータ
エンハンスメント・レイヤ1 mdat:メディア・データ・ボックス
エンハンスメント・レイヤ1のメディア・データ

エンハンスメント・レイヤ2 moov:動画ボックス
エンハンスメント・レイヤ2のメタデータ
エンハンスメント・レイヤ2 mdat:メディア・データ・ボックス
エンハンスメント・レイヤ2のメディア・データ

図2

【図3】

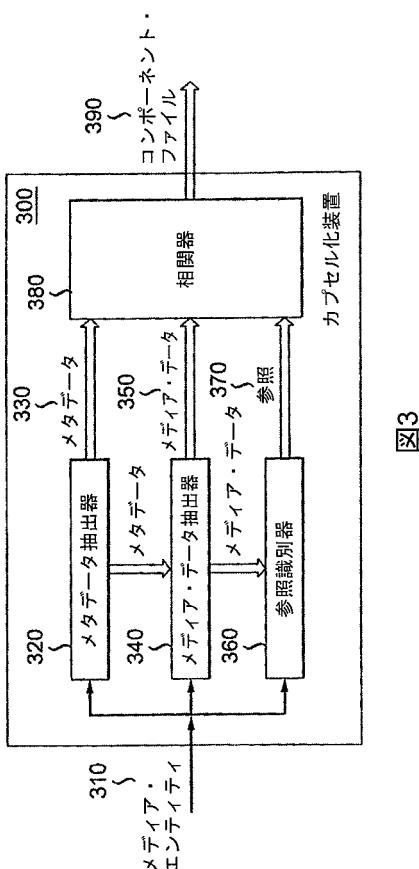


図3

【図4】

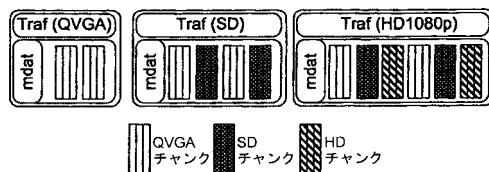


図4

【図5】

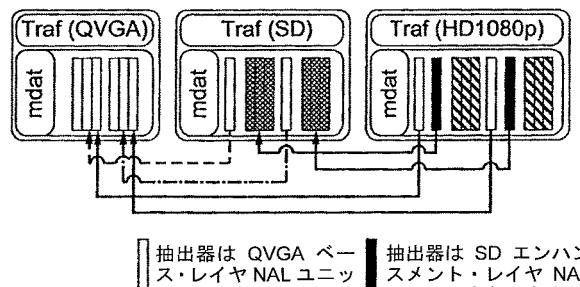


図5

【図6】

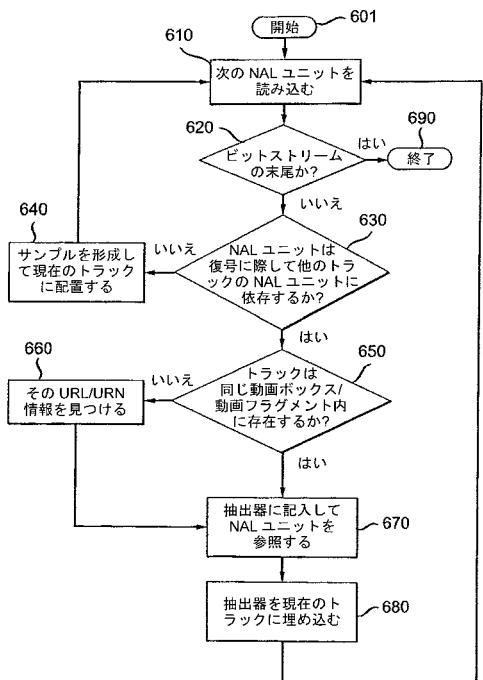


図6

【図7】

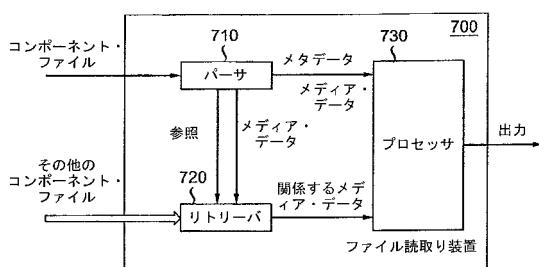


図7

【図8】

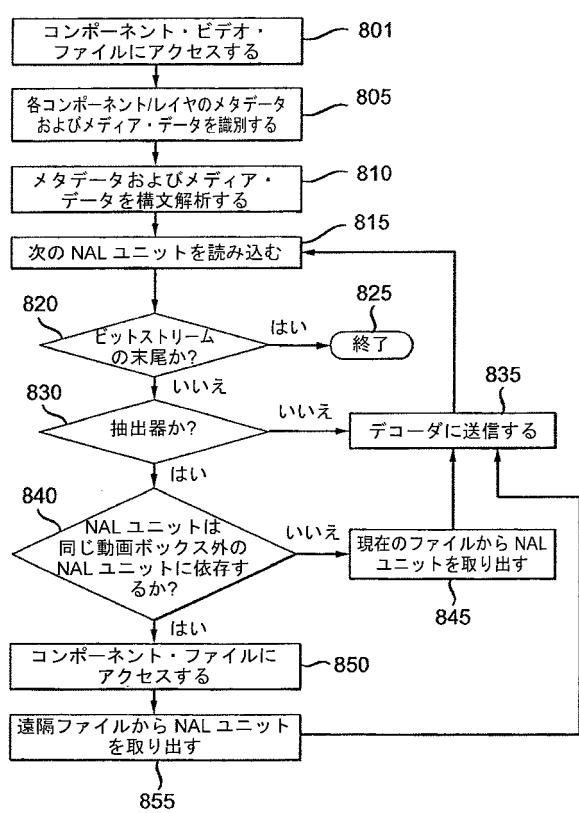


図8

【図9】

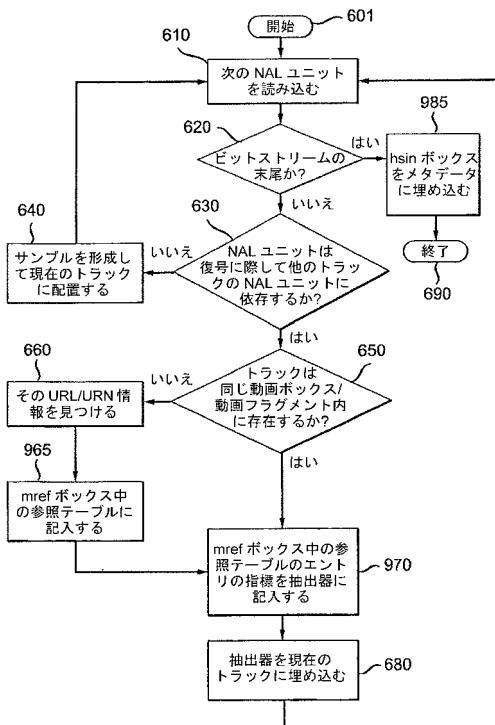


図9

【図10】

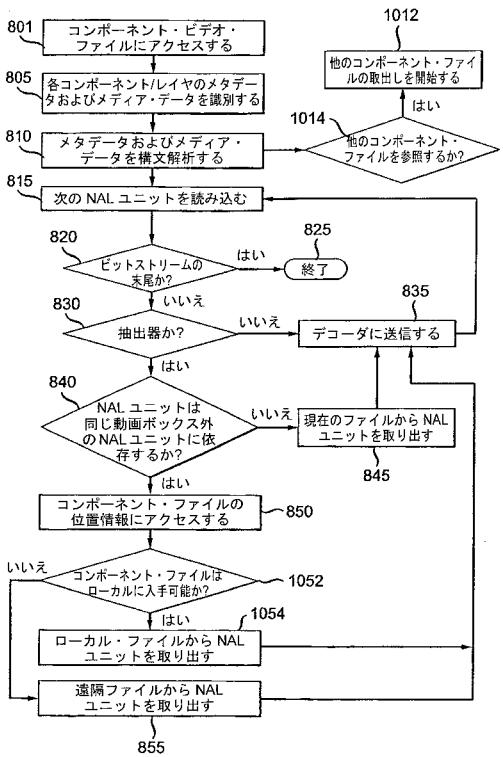


図10

【国際調査報告】

INTERNATIONAL SEARCH REPORT

		International application No PCT/US2011/040168
A. CLASSIFICATION OF SUBJECT MATTER INV. H04N21/84 H04N21/85 H04N7/24 ADD.		
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED Minimum documentation searched (classification system followed by classification symbols) H04N		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practical, search terms used) EPO-Internal, WPI Data		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category ^a	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	GRÜNEBERG K ET AL: "Deliverable D3.2 MVC/SVC storage format", no. Project No: FP7-ICT-214063 29 January 2009 (2009-01-29), pages 1-34, XP002599508, Retrieved from the Internet: URL: http://www.ist-sea.eu/Public/SEA_D3.2_HHI_FF_20090129.pdf [retrieved on 2010-09-01] section 2.2.2 and 2.2.6 figures 2-4 - 2-10 ----- - / --	1-37
<input checked="" type="checkbox"/> Further documents are listed in the continuation of Box C.		<input type="checkbox"/> See patent family annex.
* Special categories of cited documents : "A" document defining the general state of the art which is not considered to be of particular relevance "E" earlier document but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed		
Date of the actual completion of the international search 25 August 2011		Date of mailing of the international search report 31/08/2011
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016		Authorized officer Arpacı, Mutlu

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2011/040168

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X, P	ZHENYU WU ET AL: "Some extensions to ISO Base Media File Format and MPEG-2 Transport Stream to support multi-component media content HTTP Streaming", 93. MPEG MEETING; 26-7-2010 - 30-7-2010; GENEVA; (MOTION PICTURE EXPERT GROUP OR ISO/IEC JTC1/SC29/WG11),, no. M17894, 22 July 2010 (2010-07-22), XP030046484, section 1 section 2.1	1-37
A	DAVID SINGER: "Editor's draft of the Part12 file format amendment", 86. MPEG MEETING; 13-10-2008 - 17-10-2008; BUSAN; (MOTION PICTURE EXPERT GROUP OR ISO/IEC JTC1/SC29/WG11),, no. M15812, 20 October 2008 (2008-10-20), XP030044409, section 8.7.2	1-37
A	ANONYMOUS: "Text of ISO/IEC 14496-15/FDAM2 SVC File Format Extension", 83. MPEG MEETING; 14-1-2008 - 18-1-2008; ANTALYA; (MOTION PICTURE EXPERT GROUP OR ISO/IEC JTC1/SC29/WG11),, no. N9682, 12 March 2008 (2008-03-12), XP030016176, page 1, first paragraph	1-37
A	YE-KUI WANG ET AL: "Comments to the MVC file format draft", 88. MPEG MEETING; 20-4-2009 - 24-4-2009; MAUI; (MOTION PICTURE EXPERT GROUP OR ISO/IEC JTC1/SC29/WG11),, no. M16444, 17 April 2009 (2009-04-17), XP030045041, page 1, first paragraph	1-37
A	AMON P ET AL: "File Format for Scalable Video Coding", IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, IEEE SERVICE CENTER, PISCATAWAY, NJ, US, vol. 17, no. 9, 1 September 2007 (2007-09-01), pages 1174-1185, XP011193013, ISSN: 1051-8215, DOI: 10.1109/TCSVT.2007.905521 abstract	1-37

フロントページの続き

(81)指定国 AP(BW,GH,GM,KE,LR,LS,MW,MZ,NA,SD,SL,SZ,TZ,UG,ZM,ZW),EA(AM,AZ,BY,KG,KZ,MD,RU,TJ,TM),EP(AL,AT,BE,BG,CH,CY,CZ,DE,DK,EE,ES,FI,FR,GB,GR,HR,HU,IE,IS,IT,LT,LU,LV,MC,MK,MT,NL,NO,PL,PT,RO,R,S,SE,SI,SK,SM,TR),OA(BF,BJ,CF,CG,CI,CM,GA,GN,GQ,GW,ML,MR,NE,SN,TD,TG),AE,AG,AL,AM,AO,AT,AU,AZ,BA,BB,BG,BH,BR,BW,BY,BZ,CA,CH,CL,CN,CO,CR,CU,CZ,DE,DK,DM,DO,DZ,EC,EE,EG,ES,FI,GB,GD,GE,GH,GM,GT,HN,HR,HU,ID,IL,IN,IS,JP,KE,KG,KM,KN,KP,KR,KZ,LA,LC,LK,LR,LS,LT,LU,LY,MA,MD,ME,MG,MK,MN,MW,MX,MY,MZ,NA,NG,NI,NO,NZ,OM,PE,PG,PH,PL,PT,RO,RS,RU,SC,SD,SE,SG,SK,SL,SM,ST,SV,SY,TH,TJ,TM,TN,TR,TT,TZ,UA,UG,US,UZ,VC,VN,ZA,ZM,ZW

(74)代理人 100123629

弁理士 吹田 礼子

(72)発明者 ウー, チエンユ

アメリカ合衆国 ニュージャージー州 プレインズボロ タマロン・ドライブ 9016

(72)発明者 ズー, リ フア

アメリカ合衆国 カリフォルニア州 バーバンク #207 アンドーバー・ドライブ 333

F ターム(参考) 5C159 MA17 MA31 PP03 PP04 RE11 SS08 SS09 SS16 TA73 TB12

TB17 TC45 TC47 UA02 UA05

5C164 MB11S SC03P UB10P