(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2004-504739
(P2004-504739A)
(43)公表日　平成16年2月12日(2004.2.12)

(51)Int.Cl.[7]	F I		テーマコード（参考）
H04N 7/30	H04N 7/133	Z	5C059

審査請求 有　予備審査請求 有　（全 98 頁）

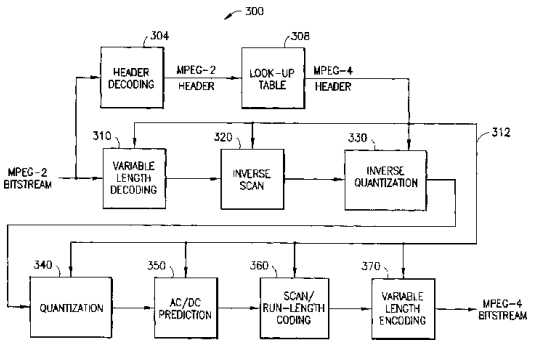
(21)出願番号	特願2002-503040(P2002-503040)
(86)(22)出願日	平成13年5月25日(2001.5.25)
(85)翻訳文提出日	平成14年12月6日(2002.12.6)
(86)国際出願番号	PCT/US2001/040811
(87)国際公開番号	WO2001/095633
(87)国際公開日	平成13年12月13日(2001.12.13)
(31)優先権主張番号	09/590,928
(32)優先日	平成12年6月9日(2000.6.9)
(33)優先権主張国	米国(US)

(71)出願人　598045380
ジェネラル・インスツルメント・コーポレイション
アメリカ合衆国ペンシルベニア州ホースハム，トーナメント・ドライブ　101
101 Tournament Drive Horsham, Pennsylvania, The United States of America
(74)代理人　100069899
弁理士　竹内　澄夫
(74)代理人　100096725
弁理士　堀　明▲ひこ▼

最終頁に続く

(54)【発明の名称】ＭＰＥＧ－２からＭＰＥＧ－４へのビデオサイズの変換およびトランスコード化

(57)【要約】
【解決手段】ＭＰＥＧ－２ビットストリームをＭＰＥＧ－４ビットストリームに変換するトランスコーダアーキテクチャーは，エラーが些細で複雑さが顕著に低い。トランスコーダは入力ビットストリームからヘッダ情報を読み取り，出力ビットストリームに対して新しいフォーマットで対応するヘッダを与える。複雑さが低い前後トランスコーダが動き補償処理の必要性を回避する。またドリフトエラーを最小化するトランスコーダが提供される。他の実施例ではサイズトランスコーダが提供され，ＭＰＥＧ－２ ＭＰ@ ＭＬでコード化する，ＩＴＵ－Ｒ６０１インターレース化されたビデオのビットストリームが，ストリーミングビデオの場合に適したＳＩＦプログレッシブビデオを含む，単純なプロファイルＭＰＥＧ－４ビットストリームに変換される。フィールドモードＤＣＴブロックの空間ダウンスケール化に対して，垂直および水平ダウンスケール化技術が計算の減少化のために，疎マトリクスを使用して組み合わされる。



【特許請求の範囲】
【請求項１】
第一のビデオコード化フォーマットで与えられる，予め圧縮された入力ビットストリームをトランスコード化する方法であって，
入力ビットストリームのヘッダ情報を修復する工程と，
第二の，異なるビデオコード化フォーマットで，対応するヘッダ情報を与える工程と，
部分的に圧縮が解かれたデータを与えるために，入力ビットストリームの圧縮を部分的に解く工程と，
出力ビットストリームを与えるために，第二のフォーマットでヘッダ情報にしたがって，部分的に圧縮が解かれたデータを再度圧縮する工程と，
を含む方法。
【請求項２】
第一および第二のビデオコード化フォーマットは，それぞれＭＰＥＧ－２フォーマットおよびＭＰＥＧ－４フォーマットからなる，請求項１に記載の方法。
【請求項３】
第一のビデオコード化フォーマットは，メインレベルの主プロファイルからなり，
第二のビデオコード化フォーマットは標準中間フォーマット（ＳＩＦ）プログレッシブ順次ビデオをもつ単純なプロファイルＭＰＥＧ－４ビットストリームからなる，請求項１に記載の方法。
【請求項４】
部分的に圧縮が解かれたデータは，動きベクトルおよび離散コサイン変換（ＤＣＴ）係数を含み，
第二のフォーマットは，新しいモード決定，ＡＣ／ＤＣ予想，および動き補償の少なくとも一つを含む，請求項１に記載の方法。
【請求項５】
少なくとも一つのルックアップテーブルが第二のビデオコード化フォーマットで，対応するヘッダ情報を与えるために使用される，請求項１に記載の方法。
【請求項６】
ダウンスケール化は，ＤＣＴ係数およびその動きベクトルをダウンサンプル化することにより，部分的に圧縮が解かれたデータについて実行される，請求項１に記載の方法。
【請求項７】
２：１のダウンスケール化は，部分的に圧縮が解かれたデータの，四つのフィールドモード離散コサイン変換（ＤＣＴ）ブロックの少なくとも一つのグループについて，二つのフレームモードＤＣＴブロックの対応するグループを得るために，垂直ダウンサンプル化およびデインターレース化を実行することにより，そして一つのフレームモードＤＣＴブロックを得るために，二つのフレームモードＤＣＴブロックに水平ダウンサンプル化を実行することにより，実行される，請求項１に記載の方法。
【請求項８】
垂直ダウンサンプル化はまた，四つのフィールドモードＤＣＴブロックを，ローパスフィルタにかけることも実施する，請求項７に記載の方法。
【請求項９】
垂直及び水平ダウンサンプル化はそれぞれ，疎マトリクスを使用する，請求項７に記載の方法。
【請求項１０】
再度圧縮を行う工程で，量子化器において，変化を特定するコード（ＤＱＵＡＮＴ）が，部分的に圧縮が解かれたデータの量子化パラメータの異なる値にしたがってセットされる，請求項１に記載の方法。
【請求項１１】
内コード化されたマクロブロックを再度圧縮するために，コード化されたブロックパターン（ＣＢＰ）が，部分的に圧縮が解かれたデータの対応する値にしたがってセットされる

，請求項１に記載の方法。
【請求項１２】
非内コード化されたマクロブロックを再度圧縮するために，部分的に圧縮が解かれたデータの，スキップされたマクロブロックが，ｎｏｔ_ｃｏｄｅｄ ｍａｃｒｏｂｌｏｃｋとしてコード化され，ここで，すべての離散コサイン変換（ＤＣＴ）係数がゼロの値をもつ，請求項１に記載の方法。
【請求項１３】
再度圧縮する工程において，部分的に圧縮が解かれたデータの予想された動きベクトルが，第二のフォーマットにしたがってリセットされる，請求項１に記載の方法。
【請求項１４】
部分的に圧縮が解かれたデータの，二重の主モードがフィールドコード化されたマクロブロックに変換される，請求項１に記載の方法。
【請求項１５】
ビデオデータに２：１のダウンスケール化を実施する方法であって，
ビデオデータから，Ｎ×Ｎ離散コサイン変換（ＤＣＴ）係数の，少なくとも一つの入力マトリクスを，四つのＮ／２×Ｎ／２フィールドモードＤＣＴブロックを組み合わせることにより，形成する工程と，
二つのＮ／２×Ｎ／２フレームモードブロックを得るために，入マトリクスに，垂直ダウンサンプル化およびデインターレース化を実施する工程と，
二つのフレームモードＤＣＴブロックから，Ｎ×Ｎ／２入力マトリクスを形成する工程と，
一つのＮ／２×Ｎ／２フレームモードＤＣＴブロックを得るために，Ｎ×Ｎ／２マトリクスに，水平ダウンサンプル化を実施する工程と，
を含む方法。
【請求項１６】
Ｎ＝１６である，請求項１５に記載の方法。
【請求項１７】
垂直ダウンサンプル化はまた，Ｎ×Ｎ入力マトリクスをローパスフィルタにかける，請求項１５に記載の方法。
【請求項１８】
垂直ダウンサンプル化は，疎ダウンサンプル化マトリクスを使用する，請求項１５に記載の方法。
【請求項１９】
疎ダウンサンプル化マトリクス＝０．５［I_8 I_8］であり，I_8は８×８単位マトリクスである，請求項１８に記載の方法。
【請求項２０】
水平ダウンサンプル化は，奇“Ｏ”および偶“Ｅ”マトリクスからなる疎ダウンサンプル化マトリクスを使用する，請求項１５に記載の方法。
【請求項２１】
偶マトリクスは
Ｅ＝［ｅ（０）　０　０　０
０　ｅ（１）　０　ｅ（２）
０　０　０　０
０　ｅ（３）　０　ｅ（４）
０　０　ｅ（５）　０
０　ｅ（６）　０　ｅ（７）
０　０　０　０
０　ｅ（８）　０　ｅ（９）］
を有し，ここでｅ（１）からｅ（９）はゼロでない係数であり，
奇マトリクスは，

Ｏ＝［０　　０　　０　　０
ｏ（０）　　０　　ｏ（１）　　０
０　　ｏ（２）　　０　　０
ｏ（３）　　０　　ｏ（４）　　０
０　　０　　０　　０
ｏ（５）　　０　　ｏ（６）　　０
０　　０　　０　　ｏ（７）
ｏ（８）　　０　　ｏ（９）　　０］
を有し，ここでｏ（１）からｏ（９）はゼロでない係数である，
請求項２０に記載の方法。

【請求項２２】
第一のビデオコード化フォーマットで与えられる，予め圧縮されたビットストリームをトランスコード化する装置であって，
入力ビットストリームのヘッダ情報を修復する手段と，
第二の，異なるビデオコード化フォーマットで，対応するヘッダ情報を与える手段と，
部分的に圧縮が解かれたデータを与えるために，入力ビットストリームの圧縮を部分的に解く手段と，
出力ビットストリームを与えるために，第二のフォーマットでヘッダ情報にしたがって，部分的に圧縮が解かれたデータを再度圧縮する手段と，
を含む装置。

【請求項２３】
ビデオデータに２：１のダウンスケール化を実施する装置であって，
ビデオデータから，Ｎ×Ｎ離散コサイン変換（ＤＣＴ）係数の，少なくとも一つの入力マトリクスを，四つのＮ／２×Ｎ／２フィールドモードＤＣＴブロックを組み合わせることにより，形成する手段と，
二つのＮ／２×Ｎ／２フレームモードブロックを得るために，入マトリクスに，垂直ダウンサンプル化およびデインターレース化を実施する手段と，
二つのフレームモードＤＣＴブロックから，Ｎ×Ｎ／２入力マトリクスを形成する手段と，
一つのＮ／２×Ｎ／２フレームモードＤＣＴブロックを得るために，Ｎ×Ｎ／２マトリクスに，水平ダウンサンプル化を実施する手段と，
を含む装置。

【発明の詳細な説明】
【０００１】
発明の背景
本発明は，マルチメディアデータの圧縮に関し，特に，一般的なＭＰＥＧ－４をＭＰＥＧ－２にデコード化することのできるビデオ・トランスコーダに関する。時間的および空間的サイズ変換（ダウンサイズ化）もまた提供される。

【０００２】
以下の頭文字および用語が使用される。
ＣＢＰ－コード化されたブロックパターン
ＤＣＴ－離散コサイン変換
ＤＴＶ－デジタルテレビ
ＤＶＤ－デジタルビデオディスク
ＨＤＴＶ－高精細度テレビ
ＦＬＣ－固定長コード化
ＩＰ－インターネットプロトコル
ＭＢ－マクロブロック
ＭＥ－動き推定
ＭＬ－メインレベル

ＭＰ－メインプロファイル
ＭＰＳ－ＭＰＥＧ－２プログラムストリーム
ＭＴＳ－ＭＰＥＧ－２トランスポートストリーム
ＭＶ－動きベクトル
ＱＰ－量子化パラメータ
ＰＭＶ－予想動きベクトル
ＲＴＰ－リアルタイムトランスポートプロトコル（ＲＦＣ１８８９）
ＳＤＴＶ－標準的精細度テレビ
ＳＩＦ－標準的中間フォーマット
ＳＶＣＤ－超ビデオコンパクトディスク
ＶＬＣ－可変長コード化
ＶＬＤ－可変長デコード化
ＶＯＰ－ビデオオブジェクトプレーン

【０００３】
ＭＰＥＧ－４（マルチメディアコード化標準）が，メディアのストリーミング，広告，双方ゲーム，仮想旅行などといったインターネットの分野の例を含む種々の応用例をサポートするための豊かな機能性を与える。インターネットによりビデオのストリーミング（マルチキャスト）（インターネットでの一般的な応用例において，期待されている）は，ＭＰＥＧ－４ビジュアル標準（国際標準（ＭＰＥＧ－４）のＩＳＯ／ＩＥＣ１４４９８－２最終ドラフト，“情報技術－オーディオビジュアル化オブジェクトの一般コード化，パート２：ビジュアル”，１９９８年１２月）とともに使用に非常に適している。

【０００４】
ＭＰＥＧ－４ビジュアルは合成ビデオおよび生のビデオの両方を取り扱い，ビデオ，フェース，およびメッシュオブジェクトのような，いくつかビジュアルオブジェクトタイプを受け入れる。ＭＰＥＧ－４はまた，多数のオブジェクトがユーザーにより望まれるとき，シーンにおいて示されまたは操作され得るように，任意に形状付けられたオブジェクトのコード化を可能にする。さらに，ＭＰＥＧ－４ビジュアルは，多数の予備的な（アルファ）プレーン，可変なフレームレート，および幾何学的なトランスフォーメーション（スプライト）のような強化された特徴を含むことにより，コード化および表示配列の点に関して非常に柔軟である。

【０００５】
しかし，ストリーミングビデオの対象である期待されるビデオ題材（たとえば，映画，スポーツ番組，コンサートなど）の大半は，ＭＰＥＧ－２により圧縮され，ＤＶＤ，コンピュータメモリー（たとえば，サーバーハードディスク）のような記憶媒体に記憶されている。ＭＰＥＧ－２システム仕様（“情報技術－動画および関連したオーディオの一般的なコード化：パート２－ビデオ”，１９９５）は，二つのシステムストリームフォーマット（ＭＰＥＧ－２トランスポートシステム（ＭＴＳ）およびＭＰＥＧ－２プログラムストリーム（ＭＰＳ））を定義する。ＭＴＳは，ＭＰＥＧ－２で圧縮されたデータの一つ以上のプログラム，またエラーの生じやすい環境での他のデータの通信または記憶のために作られる。ＭＴＳの典型的な応用例の一つがＤＴＶである。一般的な応用例はＤＶＤおよびＳＶＣＤを含む。

【０００６】
この課題を解決する試みはいまのところ満足のいくものとはなっていない。たとえば，ＭＰＥＧ－４スタジオプロファイルは（Ｏ．Ｓｕｎｏｈａｒａ，Ｙ．Ｙａｇａｓａｋｉらの“ＭＰＥＧ－４ Ｓｔｕｄｉｏ Ｐｒｏｆｉｌｅ Ａｍｅｎｄｍｅｎｔ Ｗｏｒｋｉｎｇ Ｄｒａｆｔ ２．０のドラフト”，ＩＳＯ／ＩＥＣ ＪＴＣ／ＳＣ２９ ９９／５１３５）は，ＭＰＥＧ－２からＭＰＥＧ－４のトランスコーダを提案しているが，そのようなプロセスは他のＭＰＥＧ－４バージョン１プロファイル（ナチュラルビジュアルプロファイル（Ｓｉｍｐｌｅ，Ｓｉｍｐｌｅ Ｓｃａｌｅａｂｌｅ，Ｃｏｒｅ，Ｍａｉｎ，Ｎ－Ｂｉｔ），合成ビジュアルプロファイル（Ｓｃａｌｅａｂｌｅ Ｔｅｘｔｕｒｅ，Ｓｉｍｐｌｅ

Ｆａｃｅ Ａｎｉｍａｔｉｏｎ），および合成／ナチュラルハイブリッドビジュアル（Ｈｙｂｒｉｄ，Ｂａｓｉｃ Ａｎｉｍａｔｅｄ Ｔｅｘｔｕｒｅ））に適用できない。スタジオプロファイルは，シンタックスを修正することから，ＭＰＥＧ－４バージョン１のメインプロファイルに適用できず，デコーダプロセスは残りのＭＰＥＧ－４バージョン１と互換性がない。

【０００７】

ＭＰＥＧ標準は，二次元のランキング順序を使用して，限定パラメータのいくつかのセットを指定する。“プロファイル”シリーズといわれている次元の一つが，サポートされたコード化の特徴を特定する。“レベル”といわれる他の次元は，画像の解像度，ビットレートなど（受け入れられるもの）を特定する。

【０００８】

ＭＰＥＧ－２に関し，メインレベルでのメインプロファイル，ＭＰ@ＭＬは，４：２：０のサブサンプル化比，ならびにＩ，ＰおよびＢピクチャーをサポートする。メインレベルはＩＴＵ－Ｒ６０１ビデオに対して定義される一方，シンプルレベルはＳＩＦビデオに対して定義される。

【０００９】

同様に，シンプルプロフィルはＳＩＦプログレッシブビデオ（Ｂ－ＶＯＰまたはインターレース化されたビデオをもたない）を含む。メインプロファイルはＢ－ＶＯＰおよびインターレース化されたビデオを可能にする。

【００１０】

したがって，ＭＰＥＧ－２ビデオからＭＰＥＧ－４ビデオへのトランスコード化および／またはＭＰＥＧ－４ビデオからＭＰＥＧ－２ビデオへのトランスコード化を使用することにより，端末システムの異なるタイプの間で相互運用ができるようにすることが望まれている。適用させられるべき端末システムの異なるタイプはつぎのものを含む。

【００１１】

送信相互作用ユニット（ＴＩＵ）：在来のＭＴＳ（またはＭＰＳ）からＭＰＥＧ－２ビデオを受信し，ＭＰＥＧ－４にトランスコード化し，在来のＲＴＰベースのシステムそう（ＩＰベースの相互接続ネットワークのようなもの）を使用して，パケットネットワークにより配給する。たとえば，リアルタイムエンコーダ，インターネットへのＭＴＳ衛星リンク，およびＭＰＤエンコード化したソースマテリアルをもつビデオサーバがある。

【００１２】

受信相互作用ユニット（ＲＩＵ）：ＲＴＰベースのネットワークからリアルタイムでＭＰＥＧ－４を受信し，つぎに（必要であれば）ＭＰＥＧ－２ビデオにトランスコード化し，そして在来のＭＴＳ（またはＭＰＳ）環境に送る。たとえば，インターネットベースのビデオサーバからＭＴＳベースのケーブル配給施設までである。

【００１３】

送信インターネット端末システム（ＴＩＥＳ）：インターネット端末システム自ら発生され，もしくは記憶された，またはインターネットベースのコンピュータネットワークから受信されたＭＰＥＧ－２またはＭＰＥＧ－４を送信する。たとえば，ビデオサーバがある。

【００１４】

受信インターネット端末システム（ＲＩＥＳ）：インターネット端末システムでの消費のために，ＲＴＰベースのインターネットにより，ＭＰＥＧ－２またはＭＰＥＧ－４を受信し，伝統的なコンピュータネットワークへと送る。たとえば，トレーニングビデオをみるデスクトップＰＣまたはワークステーションがある。

【００１５】

ＭＰＥＧ－２とＭＰＥＧ－４の間の類似性および相違性を決定し，複雑さが低く，エラーが些細であるトランスコーダアーキテクチャーを提供することが望ましい。

【００１６】

トランスコーダアーキテクチャーは，Ｂフレームが有効であるシステムに提供され，さら

にＢフレームが使用できないときのための（単純なプロファイル）単純化されたアーキテクチャーであるべきである。
【００１７】
フォーマット（ＭＰＥＧ－２からＭＰＥＧ－４）および／またはサイズのトランスコード化が提供されるべきである。
【００１８】
ヘッダーのマッピングを含む，ＭＰＥＧ－２からＭＰＥＧ－４シンタックスの有効なマッピングを提供することは望ましい。
【００１９】
システムは，空間および時間トランスコード化を含む，サイズのトランスコード化を含むべきである。
【００２０】
システムは，トランスコーダの入力ビットストリームまたは出力ビットストリームで，サイズの変換を可能にすべきである。
【００２１】
サイズトランスコーダは，ＭＰＥＧ－２ ＭＰ@ＭＬでコード化され，ＩＴＵ－Ｒ院てーレース化されたビデオのビットストリームを，たとえばストリーミングビデオの応用例に適したＳＩＦプログレッシブビデオを含む，単純なプロファイルのＭＰＥＧ－４ビットストリームに変換する。
【００２２】
システムは，ストリーミングビデオの応用例に対して，実際的な帯域内に適応する出力ビットストリームを提供すべきである。
【００２３】
本発明は上記および他の利点を有するシステムを提供する。
【００２４】
発明の要約
本発明は，フォーマットのトランスコード化（ＭＰＥＧ－２からＭＰＥＧ－４へ）およびサイズ（空間および時間）のトランスコード化に関する。
【００２５】
提案のトランスコーダは，サイズの変換を含むが，これらパラメータは，入力ビットストリームまたは出力ビットストリームでトランスコード化される。しかし，トランスコーダが，互いに処理要素を共有することから（ビットストリームリーダのように），複雑さを軽減するために，トランスコーダの製品バージョンに，あらゆる種類のトランスコード化を含めることが有効である。
【００２６】
本発明は，トランスコーダに対して最も重要な条件，たとえば，システムの複雑さ，およびプロセスで生じた損失に関連する。
【００２７】
一実施例において，提案の前後トランスコーダアーキテクチャーが，動き補償を達成する必要がないことから，複雑さを軽減する。
【００２８】
特定の実施例においては，トランスコーダは，可変５ビットＱＰ表示を使用し，ＡＣ／ＤＣ予想および非線形ＤＣスケーラを除去する。
【００２９】
本発明は，これに代えてレート制御およびサイズの再変更に有効である。
【００３０】
第一のビデオコード化フォーマットで与えられる，予め圧縮された入力ビットストリームをトランスコード化する特別な方法は，対応するヘッダ情報を，第二の異なるビデオコード化フォーマットで与え，入力ビットストリームの圧縮を部分的に解き，部分的に圧縮が解かれたデータを与え，出力ビットストリームを与えるために，第二のフォーマットで，

ヘッダ情報にしたがって，部分的に圧縮が解かれたデータを，再度圧縮することを含む。
【００３１】
ビデオデータを２：１でダウンスケール化する方法は，ビデオデータから，Ｎ×Ｎ（たとえばＮ＝１６）の離散コサイン変換（ＤＣＴ）計数の，少なくとも一つの入力マトリクスを，四つのＮ／２×Ｎ／２フィールドモードＤＣＴブロックを組み合わせることにより形成し，二つのＮ／２×Ｎ／２フレームモードＤＣＴブロックを得るために，入力マトリクスの，垂直方向ダウンサンプル化およびデインターレース化を実行し，二つのフレームモードＤＣＴブロックから，Ｎ×Ｎ／２入力マトリクスを形成し，一つのＮ／２×Ｎ／２フレームモードＤＣＴブロックを得るために，Ｎ×Ｎ／２マトリクスの水平ダウンサンプル化を実行することを含む。
【００３２】
好適に，垂直および水平ダウンサンプル化は，それぞれの疎ダウンサンプル化マトリクスを使用する。特に，０．５［I_8 I_8］垂直ダウンサンプル化が使用できるが，このI_8は８×８の単位マトリクスである。これは，基本的に，垂直画素の平均化である。奇“Ｏ”および偶“Ｅ”の複合した，水平ダウンサンプル化も使用できる。
【００３３】
対応する装置もまた提供される。
【００３４】
本発明は，フォーマットのトランスコード化（ＭＰＥＧ－２からＭＰＥＧ－４）およびサイズ（空間および時間）のトランスコード化に関する。
【００３５】
本発明は，予め圧縮されたビットストリームを，異なるビットレートで，他の圧縮されたビットストリームに，ビットレートのトランスコード化することを提供する。
【００３６】
ビットレートのトランスコード化は，たとえばビデオのストリーミング（ネットワークの帯域は一定でなく，ときに，ビデオサーバはネットワークトラフィック要求に対処するために，ビットレートの減縮を必要とするからである）には重要である。入力ビットストリームからのＭＶを再利用し，したがって動き推定（ＭＥ）を除去する，カスケード式トランスコーダは，最も効率的なビットレートトランスコーダである。カスケード式トランスコーダは，ＭＶを得て，そして参照フレームを形成するために，入力ビットストリームをデコード化する。つぎに，出力ビットストリームを所望のビットレートで生成するために，レート制御機構で，この情報をコード化する。
【００３７】
空間解像度のトランスコード化は，近い将来に，ＨＤＴＶとＳＤＴＶの共存にともなう大きな問題となる。インターネットの帯域が，放送用の品質をもつビデオ用に十分大きくなりそうもないことから，ビデオのストリーミングも非常に有益である。したがって，放送用の品質をもつビットストリームを，処理しやす解像度でダウンサンプル化することは必要なことである。空間解像度のトランスコード化は，システムの複雑を顕著に減少させることから，圧縮された（ＤＣＴ）ドメインで実行される。圧縮されたドメインでのダウンサンプル化のプロセスは，二つのパラメータ，すなわちＤＣＴ係数およびＭＶの処理に関する。ダウンサンプル化フィルタおよびその高速アルゴリズムがＤＣＴ係数ダウンサンプル化を実行するために，提案されている。実際の製品で，ドリフトを回避するために，動き補償の残差は，入力ビットストリームからのＤＣＴ係数を近似化することに代え，再度変換されなければならない。
【００３８】
２．　高レベルの比較
構成，ＭＰＥＧ－２およびＭＰＥＧ－４は同様の圧縮アルゴリズムを利用する。根本的に，両方の標準は，時間相関を活用するために動き予想を採用し，フレーム内で空間相関を使用するために，ＤＣＴドメインにおける量子化を用いる。ここでは，高レベルのＭＰＥＧ－２およびＭＰＥＧ－４デコーダの構成を説明，二つの標準の間の違いを示す。

【００３９】
２．１　ＭＰＥＧ－２
図１はＭＰＥＧ－２の，単純化されたビデオデコード化プロセスを示す。デコーダ１００において，コード化されたビデオデータは，一次元のデータＱＦＳ［ｎ］（ここでｎは０から６３の範囲の係数指数である）を与えるために，可変長デコード化機能１１０に与えられる。逆走査機能１２０で，ＱＦＳ［ｎ］はＱＦ［ｖ］［ｕ］（ここで，アレーインデックスｕおよびｖは０から７の範囲にともにある）で示された係数の二次元アレーに変換される。逆量子化機能１３０が，再構成された，最終的な周波数ドメインのＤＣＴ係数Ｆ［ｖ］［ｕ］を与えるために，適切な逆量子化の計算を行う。逆ＤＣＴ機能１４０が，画素（空間）ドメイン値ｆ［ｙ］［ｘ］を発生する。動き補償機能１５０が，デコード化された画素ｄ［ｙ］［ｘ］を発生するために，フレーム記憶メモリー１６０および値ｆ［ｙ］［ｘ］（ここでｙおよびｘは画素ドメインにおけるデカルト座標である）に応答する。
【００４０】
ＭＰＥＧ－２は，動き補償に対してマクロブロックレベルに，ＤＣＴ変換に対してブロックレベルに，ランレングスおよび損失のないコード化に対して係数レベルに作用する。さらに，ＭＰＥＧ－２は，三つのタイプのピクチャー，すなわち，Ｉピクチャー，Ｐピクチャーおよびピクチャーを可能にしている。可能になった動き補償モード（前方，後方，両方向）はＰピクチャーおよびＢピクチャーに対して特定される。ＭＰＥＧ－２はインターレース化されたソースをより効率的に扱うために，インターレースされたコード化ツールを使用する。
【００４１】
２．２　ＭＰＥＧ－４
図２は，拡張性のない，ＭＰＥＧ－４のビデオデコード化プロセスを示す。
【００４２】
デコーダ２００において，チャネルからのデータがデマルチプレクサ２１０から出力される。形状データの，コード化されたビットストリームが，ＭＰＥＧ－４用語，ｖｉｄｅｏ＿ｏｂｊｅｃｔ＿ｌａｙｅｒ＿ｓｈａｐｅ（たとえば，現在のイメージが矩形か，バイナリーのみか，またはグレースケールかどうかを示す）とともに，スイッチ２１５に与えられる。ｖｉｄｅｏ＿ｏｂｊｅｃｔ＿ｌａｙｅｒ＿ｓｈａｐｅが“００”に等しいと，バイナリー形状デコード化は必要とされない。そうでないとき，バイナリー形状デコード化は実行される。
【００４３】
バイナリー形状デコード化が実行されると，形状デコード化機能は，前の再構成されたＶＯＰ２３０（メモリーに記憶されてもよい）を受信し，形状デコード化された出力が動き補償機能２４０に与えられる。動き補償機能２４０は動き補償機能２３５（順に，デマルチプレクサ２１０から，コード化された動きビットストリームを受信する）から出力を受信する。動き補償機能２４０はまた，ＶＯＰ再構成機能２４５に出力を与えるために，再構成された前のＶＯＰ２３０を受信する。
【００４４】
ＶＯＰ再構成機能２４５はまた，テクスチャーコード化機能２５０（順に，形状デコード化機能２２０からの出力に加え，デマルチプレクサ２１０から，テクスチャーコード化されたビットデータを受信する）からデータを受信する。テクスチャーデコード化機能２５０は，可変長デコード化機能２５５，逆走査機能２６０，逆ＤＣおよびＡＣ予想機能２７０，逆量子化機能２８０およびＩｎｖｅｒｓｅ（逆）ＤＣＴ機能２９０を含む。
【００４５】
ＭＰＥＧ－２と比較して，いくつかの新しいツールが，特徴および双方方向性，たとえば，スプライトコード化，形状コード化，テクスチャーコード化，拡張性，およびエラー耐性を付加するために，ＭＰＥＧ－４において用いられる。さらに，ＭＰＥＧ－４における動き補償およびテクスチャーコード化ツール（ＭＰＥＧ－２のビデオコード化に似ている）は，コード化の効率，たとえば，直接モードの動き補償，無制限動き補償，および最新

予想のようなコードツールを改良するために，修正される。
【００４６】
特に，直接モードの動き補償は，Ｂ－ＶＯＰに対して使用される。とりわけ，ＩまたはＰ－ＶＯＰマクロブロックＭＶを用い，そして，Ｂ－ＶＯＰのマイクロブロックに対してＭＶを進め，後退するように，それらを推定することにより導かれた両方向モード補償を使用する。実際のＭＶは，デルタベクトルおよびその共同で位置を占めるマクロブロックからの，スケール化されたＭＶから計算される。
【００４７】
無制限動き補償は，マクロブロック当たり一つまたは四つのＭＶを可能にする。四つのＭＶモードは直接モードの使用で，Ｂ－ＶＯＰにおいてのみ可能である。クロミナンスマクロブロックに対するＭＶは，関連したルミナンスマクロブロックからの四つのＭＶの平均であることに留意されたい。さらに，無制限動き補償は，ＭＶが参照フレーム（境界の外のテクスチャーがエッジ画素で補充される）から指示することを可能にする。
【００４８】
最新予想は，ＭＶおよびＤＣＴ係数に対し予想方法を定義する。ＭＶ予想子が三つの隣接したＭＶの中央値にしたがってセットされる。内ＤＣＴ係数の予想は，内ＡＣ／ＤＣ予想手順（グラハム則）に従う。
【００４９】
３．　トランスコーダアーキテクチャー
図３は，Ｂフレームは無効とされた，本発明にしたがう複雑性の低い前後トランスコーダを示す。
【００５０】
ＭＰＥＧ－２とＭＰＥＧ－４との構成の間の類似点は，複雑性の低い（前後）トランスコーダを可能にすることである。ＭＰＥＧ－２のビットストリームを空間（画素）ドメインレベルに完全にデコード化することに代えて，前後トランスコーダ３００は，動き推定プロセスを実際に実行することなく，ＭＰＥＧ－４のビットストリームを発生するために，ＤＣＴ係数およびＭＶを使用する。このアーキテクチャーは，再構成されたフレームにドリフトを生じさせるが，ビットレートの制御を不可能にするという，トレードオフの関係がある。しかし，ドリフトの問題は，ＭＰＥＧ－２とＭＥＰＧ－４デコーダの間の違いのほとんどが，損失のないコード化部分にあることから，些細なものでる。
【００５１】
トランスコーダ３０は，ヘッダデコード化機能３０４，ルックアップテーブル３０８および通信経路３１２とともに，ＭＰＥＧ－２ビットストリームリーダ（デコーダ）（３１０－３３０）およびＭＰＥＧ－４ヘッダ，およびテクスチャーコーダ（エンコーダ）（３４０－３７０）のカスケードからなる。トランスコーダ３００は入力ＭＰＥＧ－２ビットストリームを読み取り，機能３１０で，ＤＣＴ係数およびＭＶ残差に基づいて，可変長デコード化（ＶＬＤ）を実行し，つぎにフレーム内の，すべてのブロックのＤＣＴ係数および／またはＭＶを見いだすために，ＭＰＥＧ－２論理にしたがっていく。
【００５２】
ヘッダデコード化機能３０４はＭＰＥＧ－２ヘッダをデコード化し，それらをルックアップテーブル（または類似した機能）（対応するＭＰＥＧ－４ヘッダを得るために，以下詳説するテーブルを使用する）に与える。
【００５３】
ヘッダ，ＤＣＴ係数および／またはＭＶの情報に関して，トランスコーダ３００はこの情報をＭＰＥＧ－４フォーマットにエンコード化する。参照フレームがこのアーキテクチャーでは不要であることに留意されたい。
【００５４】
トランスコーダ３００は，入力ビットストリームからＭＰＥＧ－４ヘッダを読み取り，対応するＭＰＥＧ－４ヘッダを出力ビットストリームの所定の所に書き込む。
【００５５】

ＶＬＤ３１０での処理の後，データは，逆走査機能３１０，そして逆量子化機能３３０に与えられる。つぎに，経路３１２を経て与えられたＭＰＥＧ－４ヘッダ情報を使用して，デコード化されたＤＣＴ係数データはＭＰＥＧ－４ヘッダ，および量子化されたＤＣＴ係数を，差分的にエンコード化するためのＡＣ／ＤＣ予想機能３５０，ならびに量子化機能３４０を含むテクスチャーコーダで処理される。特に，ＡＣ／ＤＣ予想プロセスは，ＤＣ係数，およびＡＣ係数の第一のローか第一のカラムのいずれかを差し引くことにより，内ＭＢのＤＣおよびＡＣ／ＤＣＴ係数の残差を発生させる。予想器は適用するように選択される。ＡＣ／ＤＣ予想機能３５０はＭＰＥＧ－４ヘッダ情報を必要としない。

【００５６】

続いて，走査／ランレングスコード化機能３６０および可変長エンコード化機能３７０はＭＰＥＧ－４ビットストリームを与える。

【００５７】

図４は，本発明にしたがった，Ｂフレームが有効で，ドリフトエラーを最小化するトランスコーダアーキテクチャーを示す。

【００５８】

図において，同様の要素には同じ符号が付されている。

【００５９】

再構成されたフレーム中のドリフトの問題，およびビットレート制御の不足に対応するために，トランスコーダ４００（図３のトランスコーダ３００を延長したもの）のような，より複雑なアーキテクチャーが使用され得る。このアーキテクチャーは実際に，テクスチャー／残差データのＤＣＴ係数を計算し，したがって動き補償が要求される。このトランスコーダのエンコーダは，デコード化プロセスを含むことから，ドリフトエラーは最小化することができる。

【００６０】

さらに，トランスコーダ４００は，ＰＭＥＧ－４がＢフレームに対して内モードを可能にしないことから，Ｂフレームでビットストリームをコードするために使用され得る。トランスコーダ４００は，（ＭＰＥＧ－４の）内モードのゼロＭＶをもつブロックのように，（ＭＰＥＧ－２の）内モードのブロックを処理する。それは，ＭＶがＰＭＶに対して予想してコードされることから，ゼロ残差ＭＶ（ＰＭＶ）か，ゼロＭＶ（ゼロでないＭＶモードを与える）かのいずれかである。

【００６１】

特に，トランスコーダ４００は，残差データをＭＶデコーダ４２５に与え，ＤＣＴ係数データを逆走査機能３２０に与える可変長デコード化機能４０５を含む。ＤＣＴデータは，画素ドメインデータを得るために，逆量子化機能３３０および逆ＤＣＴ機能４２０により処理される。内コード化された画素データは経路４２２を経てバッファに与えられる一方，内コード化された画素データは経路４２４を経て加算器４３５に与えられる。

【００６２】

経路４２４上の画素（差分）データは，内コード化されたデータを，経路４４８を経てバッファ４５０に与えるために，動き補償機能４３０（ＭＶデコーダ４２５に応答して）から参照画素データに加えられる。

【００６３】

たとえば，ＭＰＥＧ－４での再度エンコード化のために，バッファ４５０は，内画素データを直接，ＤＣＴ機能４５５に入力するか，間画素データを減算器４４５（ここで，（ＭＶデコーダ４２５に応答して）動き補償機能４４０からの出力に対する差分はＤＣＴ機能４５５に与えられる）に出力される。

【００６４】

ＤＣＴ係数はＤＣＴ機能４５５を量子化機能３４０に与えられ，量子化されたＤＣＴデータはＡＣ／ＤＣ（ＤＣＴ係数）予想機能３５０（ここで，現在のＭＶのＡＣ／ＤＣ残差は発生する）に与えられる。

【００６５】

ＤＣＴ係数は，ＤＣＴ機能４５５から量子化機能３４０に与えられ，量子化されたＤＣＴデータはつぎに，ＡＣ／ＤＣ（ＤＣＴ係数）予想機能３５０（ここで，現在のＭＢのＡＣおよびＤＣ残差は発生する）に与えられる。ＤＣＴ係数のこれら残差は，エントリピーコード化される。出力データは，走査／ランレングスコード化機能３６０に与えられ，その出力は，ＭＰＥＧ－４に準拠したビットストリームを得るために，可変長エンコード化機能３７０に与えられる。

【００６６】

量子化されたＤＣＴ係数はまた，量子化機能３４０から逆量子化機能４９５へ出力され，その出力は逆ＤＣＴ機能４９０に与えられ，その出力は，動き補償機能４４０の出力とともに，加算器４８５で合計される。

【００６７】

ヘッダデコード化機能３０４およびルックアップテーブル３０８，および経路３１２は，図３に関連して説明したように動作し，機能３０４－３７０で，ＭＰＥＧ－４フォーマットへの再エンコード化を制御する。

【００６８】

４．　フォーマットトランスコーダの実行

このセクションで，たとえば，上記した図３および図４，ならびに以下で説明される図５において実行されるように，フォーマットトランスコード化の実行が説明される。比較的重要でない実行（たとえば，時間スタンプ等の使用のようなシステムに関連した事柄）は，特に説明されるまでもなく，当業者には明らかである。

【００６９】

特に，本発明のトランスコーダは，重要なプロファイル，メインレベル（ＭＰ@ＭＬ）ＭＰＥＧ－２ビットストリームは，メインプロファイルＭＰＥＧ－４ビットストリームに変換するために使用することができる。ＭＰＥＢＧ－２ビットストリームはフレームピクチャー構成において，Ｂピクチャーコード化（二重の重要な予想ではない）でもってコード化されるとする。一般的に，ＭＰＥＧ－２で使用されたのと同じコード化モードが維持されなければならない。このモードは，ＭＰＥＧ－４において最適である可能性が高く，したがってモード決定プロセスの複雑さを避ける。ＭＰＥＧ－４の透明性パターンは常に１（一つのＶＯＰにおいて，同じサイズのＶＯＰをもつ一つの矩形のオブジェクト）である。つまり，ＭＰＥＧ－４は，ゼロでない透明性パターンにより定義される，任意に形状付けられたオブジェクトを可能にする。この特徴は，トランスコード化オブジェクトのすべての透明性パターンが一つであると安全に仮定できようには，ＭＰＥＧ－２内に存在しない。

【００７０】

４．１　ＭＰＥＧ－２ビットストリームリーダ

本発明にしたがったトランスコーダは，ビットストリームヘッダ，ＤＣＴ係数およびＭＶをＭＰＥＧ－２ビットストリームから得る。この情報は，ビットストリームで一緒に混合される。ＭＰＧＥ－２およびＭＰＥＧ－４ビットストリームの両方は，いくつかの層からなる階層的構造を用いる。各層は，多数のサブ層が続くヘッダで開始する。表１に示されているように，この実行について，ＭＰＥＧ－２層は，ＭＰＥＧ－２のスライス層（これはＭＰＥＧ－４では使用されない）を除き，ＭＰＥＧ－４への直接の翻訳をなす。ＭＰＥＧ－４のＤＣ係数および予想ＭＶはスライス層を開始するブロックにおいてリセットされる。

【００７１】

しかし，いくつかのＭＰＥＧ－４ヘッダはＭＰＥＧ－２と異なり，その逆もある。都合よく，ＭＰＥＧ－２およびＭＰＥＧ－４ヘッダ情報における制限は，ＭＰＥＧ－４ヘッダを特定するのに十分である。表２ないし表６は，ＭＰＥＧ－４，およびＭＰＥＧ－２へのそれらの関係または各層における制限をリストする。

【００７２】

【表１】

表１MPEG-2とMPEG-4の層の関係

MPEG-2	MPEG-4
ビデオシーケンス	ビデオオブジェクトシーケンス（VOS）/ ビデオオブジェクト
シーケンス拡張可能な延長	ビデオオブジェクト層（VOL）
ピクチャーのグループ（GOP）	ビデオオブジェクトプレーンのグループ（GOV）
ピクチャー	ビデオオブジェクトプレーン（VOP）
マクロブロック	マクロブロック

【００７３】
【表２】

表２　MPEG-4ヘッダおよびその導出（VOPおよびVO）

ヘッダ	コード	コメント
Visual_object_sequence _start_code	00001B0	ビジュアルセッションを開始
Profile_and_level_indication	00110100	メインプロファイル/レベル4
Visual_object_sequence _end_code	00001B1	ビジュアルセッションを終了
Visual_object_start_cod	00001B5	ビジュアルオブジェクトを開始
Is_visual_object_identifier	0	優先のバージョン識別が特定される必要ない。
Visual_object_type	0001	ビデオID
Visual_object_start_code	0000010X-0000011X	新しいビデオオブジェクトをマークする
Video_signal_type	MPEG-2から導出	MPEG-2sequence_display-extension _idに対応
Video_format	MPEG-2と同じ	MPEG-2sequence_display-extension _idに対応
Video_range	MPEG-2から導出	MPEG-2sequence_display-extension _idに対応
Colour_description	MPEG-2と同じ	MPEG-2sequence_display-extension _idに対応
Colour_primaries	MPEG-2と同じ	MPEG-2colour_descriptionに対応
Transfer_characteristics	MPEG-2と同じ	MPEG-2colour_descriptionに対応
Matrix_coefficients	MPEG-2と同じ	MPEG-2colour_descriptionに対応

【００７４】
【表３】

表３　MPEG-4 ヘッダおよびその導出（VOL）

ヘッダ	コード	コメント
Video_object_layer_start_code	0000012X	新しいビデオオブジェクト層をマークする
Random_accessible_vol	0	非内コード化された VOP を可能する
Video_object_type_identification	00000100	メインオブジェクトタイプ
Is_object_type_identifier	0	優先のバージョン識別は特定される必要なし
Aspect_ratio_info	MPEG-2 と同じ	MPEG-2 aspect_ratio information に対応
Par_width	MPEG-2 と同じ	MPEG-2 Vertical_size に対応
Par_height	MPEG-2 と同じ	MPEG-2 horizontal_size に対応
Vol_control_parameters	MPEG-2 と同じ	MPEG-2 extension__start__code identifier（シーケンス延長）に対応
Chroma_format	MPEG-2 と同じ	MPEG-2 chroma_format に対応
Low_delay	MPEG-2 と同じ	MPEG-2 low_delay に対応
Vbv_parameters	再計算	MPEG-4 VBV スペックに続く
Video_object_layer_shape	00	矩形
Vop_time_increment_resolution	再計算	表７を参照
Fixed_vop_rate	1	全ての VOP が固定レートでコード化されることを示す
Fixed_vop_time_increment	再計算	表７を参照
Video_object_layer_width	MPEG-2 と同じ	display_vertical_size に対応
Video_object_layer_height	MPEG-2 と同じ	display_horizontal_size に対応
interlaced	MPEG-2 と同じ	progressive_sequence に対応
Obmc_disable	1	無効の OBMC
Sprite_enable	0	スプライトの不在を示す
Not_8_bit	MPEG-2 から導出	MPEG-2 intra_dc_precision に対応
Quant_type	1	MPEG 量子化
Complexity_estimation_disable	1	無効の複雑性評価ヘッダ
Resync_marker_disable	1	resync_marker の不在を示す
Data_partitioned	0	無効のデータ分割化
Reversible_vlc	0	無効の逆転可能な vlc
Scalability	0	現在の層がベース層として使用されることを示す

【００７５】
【表４】

表４　MPEG-4 ヘッダおよびその導出（VOP）

ヘッダ	コード	コメント
Vop start_code	000001B6	ビデオオブジェクトプレーンの開始をマーク
Vop_coding_type	MPEG-2 と同じ	MPEG-2 picture_coding_type に対応
Modulo_time_base	再発生	MPEG-2 スペックに続く
Vop_time_increment	再発生	MPEG-2 スペックに続く
Vop_coded	1	続くデータが VOP に対して存在することを示す
Vop_rounding_type	0	rounding control の値を‘0’にセット
Change_conversion _ratio_disable	1	conv_ratio が全てのマクロブロックに対して‘1’とする
Vop_constant_alpha	0	ビットストリームに vop_constant _alpha_value を含まない
Intra_de_vlc_thr	0	全 VOP に対して内 DC vlc を使用
Top_field_first	MPEG-2 と同じ	MPEG-2 top_field_first に対応
Alternate_vertical_scan _flag	MPEG-2 と同じ	MPEG-2 alternate_scan に対応
Vop_quant	MPEG-2 から導出	MPEG-2 quantiser_scale_code に対応
Vop_fcode_forward	MPEG-2 と同じ	セクション 4.3 を参照
Vop_fcode_backward	MPEG-2 と同じ	セクション 4.3 を参照

【００７６】
【表５】
表５　MPEG-4 ヘッダとその導出

ヘッダ	コード	コメント
Not_coded	MPEG-2 から導出	MPEG-2 macroblock _address_increment に対応
Mcbpc	MPEG-2 から導出	MPEG-2 macroblock_type に対応
Ac_pred_flag	0	無効の内 AC 予想
Cbpy	MPEG-2 から導出	セクション 4.2 を参照
Dquant	MPEG-2 から導出	セクション 4.2 を参照
Modb	MPEG-2 から導出	macroblock_type に対応
Mb_type	MPEG-2 から導出	macroblock_type に対応
Cbpb	MPEG-2 から導出	セクション 4.2 を参照
Dbquant	MPEG-2 から導出	セクション 4.2 を参照
Horizontal_mv_data	MPEG-2 から導出	MPEG-2 motion_code[r][s][0]に対応
Vertical_mb_residual	MPEG-2 から導出	MPEG-2 motion_code[r][s][1]に対応
Horizontal_mv_residual	MPEG-2 から導出	MPEG-2 motion_residual[r][s][0]に対応
Vertical_mv_residual	MPEG-2 から導出	MPEG-2 motion_residual[r][s][1]に対応

【００７７】
【表６】

表６　MPEG-4 ヘッダおよびその導出

（ブロックおよびインターレース化された情報）

ヘッダ	コード	コメント
Dct_dc_size_luminance	MPEG-2 と同じ	MPEG-2 dct_dc_size_luminance に対応
Dct_dc_differential	MPEG-2 と同じ	dct_dc_differential に対応
Dct_dc_size_chrominance	MPEG-2 と同じ	MPEG-2 dct_dc_size_chrominance に対応
DCT_coefficient	MPEG-2 から導出	セクション 4.2 を参照
DCT_type	MPEG-2 と同じ	MPEG-2DCT_type に対応
Field_prediction	MPEG-2 と同じ	MPEG-2 frame_motion_type に対応
Forward_top_field _reference	MPEG-2 と同じ	MPEG-2 motion___vertical___field select[0][0]に対応
Forward_bottom _field_reference	MPEG-2 と同じ	MPEG-2 motion___vertical___field select[1][0]に対応
Backward_top_field _reference	MPEG-2 と同じ	MPEG-2 motion___vertical___field select[0][1]に対応
Backward_bottom _field_reference	MPEG-2 と同じ	MPEG-2 motion___vertical___field select[1][1]に対応

【００７８】

【表７】

表７　MPEG-4 の vop_time_increment_resolution および fixed_vop_time increment への，

MPEG-2 の frame_rate_code のマッピング

Frame rate code	Vop_time_increment_resolution	Fixed_vop_time_increment
0001	24,000	1001
0010	24	1
0011	25	1
0100	30,000	1001
0101	30	1
0110	50	1
0111	60,000	1001
1000	60	1

【００７９】

ＭＶデータはマクロブロック層に記憶される。各マクロブロックに対して四つまでのＭＶが可能である。さらに，ＭＶがフィールド化フレームタイプのいずれかであってもよく，全画素解像度から半分の画素解像度のいずれかをもってもよい。ＭＰＥＧ－２ＭＶデコード化プロセスはｍｏｔｉｏｎｃｏｄｅ（ＶＬＣ）およびｍｏｔｉｏｎ＿ｒｅｓｉｄｕａｌ（ＦＬＣ），したがってデルタを決定するために用いられる。予想ＭＶとの組み合わせで，デルタはフィールド／フレームＭＶを与える。スキップされたマクロブロックに対するＭＶはゼロにセットされる。

【００８０】

ＤＣＴデータはブロック層に記憶される。それは，まず，ビットストリーム（ＶＬＣ）からデコード化され，ジグザグまたは他の走査パターンを使用して逆走査され，つぎに逆量子化される。内ＤＣ係数はｄｃｔ＿ｄｃｄｉｆｆｅｒｅｎｔｉａｌおよび予想子（予想子はＭＰＥＧ－２のスペックにしたがってリセットされる）から決定される。スキップされたマクロブロックのＤＣＴ係数はゼロにセットされる。

【００８１】

４．２　テクスチャーコード化

本発明にしたがったトランスコーダはＤＣＴ係数（間フレームに対して）再使用する。以

下のガイドラインが使用されるべきである。
１．　ｑ_ｓｃａｌｅ_ｔｙｐｅ＝１がＭＰＥＧ－２量子化で使用される。
２．　ＭＰＥＧ量子化方法は，ＭＰＥＧ－２とＭＰＥＧ－４フレーム（ドリフト）再構成の不整合を減少するために，ＭＰＥＧ－４量子化においてのみ（Ｈ．２６３ではなく）使用されるべきである。
３．　ＭＰＥＧ－２の差分値がＭＰＥＧ－４のｄｑｕａｎｔを決定する。ｄｑｕａｎｔは，差分値が±２であるときは±２にセットされる。ｄｑｕａｎｔはＩおよびＰ－ＶＯＰに対して，量子化器における変化，すなわちｑｕａｎｔを特定する２ビットのコードである。
４．　量子化マトリクスは，ＭＰＥＧ－２ビットストリームのマトリクスの変化に続いて，変化されるべきである。
５．　トランスコーダは，ＶＯＬレベルで，他の垂直走査方法（インターレース化されたシーケンスに対して）を可能にする柔軟性をもつ。
６．　内ＡＣ／ＤＣ予想（現在のブロックのＱＰが予想ブロックのと同じでないときに，基準化することに関する）は，ＡＣ量子化における複雑さおよび不整合を減ずるために，マクロブロックレベルでオフにされるべきである。
７．　より高い効率が，たとえば，量子化パラメータ（ｉｎｔｒａ_ｄｃ_ｖｌｃｔｈｒが０か７（これらの閾値はＱＰに関わらず内ＤＣまたはＡＣの使用を強いるものである）のいずれかのときを除いて）の関数として，内ＤＣ係数のコード化のために，適切なＶＬＣテーブル（ＡＣ／ＤＣ）を選択すべく，ｉｎｔｒａ_ｄｃ_ｖｌｃ_ｔｈｒの使用でもって得ることができる。
８．　スキップされたマクロブロックがｎｏｔ_ｃｏｄｅｄ_ｍａｃｒｏｂｌｏｃｋ（すべてのＤＣＴ係数がゼロ）としてコード化される。
９．　ｃｂｐｙおよびｃｂｐｃ（ＣＢＰ）がｃｏｄｅ_ｂｌｏｃｋ_ｐａｔｔｅｒｎ ４２０（ＣＢＰ_４２０）にしたがってセットされる。内マクロブロックに対して，ＭＰＥＧ－４のＣＢＰとＭＰＥＧ－２のＣＢＰ_４２０との間に僅かな食い違いがあることに留意されたい。特に，ＣＢＰ_４２０がセットされると，そのブロックのＤＣＴ係数の一つがゼロでないことを示す。ＣＢＰは内マクロブロックのＤＣ係数（ｉｎｔｒａ_ｄｃ_ｖｌｃ_ｔｈｒにも依存して）に対応しないことを除き同様の情報を含む。したがって，ＣＢＰ４２０が内マクロブロックにおいてゼロであるとき（この場合はＩ－ＶＯＰおよびＰ－ＶＯＰで起きるが，Ｂ－ＶＯＰでは起きない）ＣＢＰがゼロでないことが可能となる。テクスチャーコード化，すなわち，ＤＣ量子化に対してＱＰコード化，ＤＣ予想および非線形スケーラに，三つの損失のソースがある。ＭＰＥＧ－４はＱＰをコード化するために，差分コード化を使用する。ＭＰＥＧ－２は，５ビットの負担で，全ての可能な３２個のＱＰ値を可能にする。しかし，差分値は±２（ＱＰ値単位）まで取ることができ，したがって，±２より大きな差分値が損失となる。この損出は，ＭＰＥＧ－２のレート制御アルゴリズムにおけるマクロブロック間でのＱＰ変動を限定することにより最小化させることができる。全ての内マクロブロックは適応性のあるＤＣ予想（これは，前のマクロブロック（ＭＰＥＧ－２ＤＣ予想）と異なる予想をとり，これにより量子化に対して異なるＤＣ残差を引き起こす）を実行する。ＭＰＥＧ－４の全ての内マクロブロックＤＣ係数がまた，非線形スケーラのため，ＭＰＥＧ－２と異なる方法で量子化される。したがって，ＭＰＥＧ－２およびＭＰＥＧ－４コード化に対する量子化されたＤＣ係数は内マクロブロックに対して異なるようである。

【００８２】

４．３　ＭＶコード化

トランスコーダは，ＭＶをＭＰＥＧ－４のフォーマットにエンコード化する。しかし，ＭＶコード化は損失のないプロセスであることから，ＭＶをＭＰＥＧ－２からＭＰＥＧ－４にトランスコード化することに関してエラーがない。次の制約がＭＰＥＧ－４に課せられる。
１．　制限のない動き補償モードが無効であるが，これは，フレームのバイナリーの外に

指定するＭＶがないことを意味する。
２．　最新予想モードが用いられる。異なる予想子（中央値）がＭＰＥＧ－４において使用されるが，しかし８×８画素ブロックに対するＭＶは使用されない。すなわち最新予想器は，８×８ＭＶおよび非線形（中央値フィルタ）予想子を有効にする。唯一，非線形予想子が四つのフォーマットトランスコーダにおいて使用される（１６×１６ＭＶを依然として保持する）。
３．　直接モードはＭＰＥＧ－４ビットストリームにおいては可能とならず，このことは，Ｂ－ＶＯＰに対して，四つのＭＶタイプのみ，すなわち，１６×１６前方および後方ベクトル，ならびに１６×８前方および後方フィールドベクトルがある。
４．　フィールド動き補償が，１６×８フィールドベクトルが使用されるときは（モードを維持する）適用される。
５．　スキップされたマクロブロックがｎｏｔ_ｃｏｄｅｄ　ｍａｃｒｏｂｌｏｃｋ（ゼロのＭＶをもつ動き補償）としてコード化される。
６．　一つのｆ_ｃｏｄｅがＭＰＥＧ－４において有効となる。したがって，二つの方向（垂直，水平）の間で，ＭＰＥＧ－４のより大きなｆ_ｃｏｄｅは次の関係，すなわちｆ_ｃｏｄｅ（ＭＰＥＧ－４）＝ｆ_ｃｏｄｅ（ＭＰＥＧ－２）－１に基づいてＭＰＥＧ－４のｆ_ｃｏｄｅに変換される。
７．　全体の参照フレームに対するテクスチャーが知れていることから，パッディングプロセスが使用されない。
８．　フィールド動き補償が，二重の主演算が起動するときは使用される。ベクトルのパリティー（参照のフィールドおよび予想フレームのフィールド）が保存される。フィールドＭＶは，ＭＰＥＧ－２ビットストリームにおいてコード化されるｖｅｃｔｏｒ［０］［０］［１：０］にしたがって発生する。同じパリティーの予想が使用されるとき（たとえば，トップフィールドからトップフィールド，またはボトムフィールドからボトムフィールド），両フィールドＭＶはｖｅｃｔｏｒ［０］［０］［１：０］である。奇パリティーの予想が使用されるとき（たとえば，トップフィールドからボトムフィールド，またはボトムフィールドからトップフィールド），トップフィールドＭＶはｖｅｃｔｏｒ［２］［０］［１：０］を使用し，ボトムフィールドＭＶはｖｅｃｔｏｒ［３］［０］［１：０］を使用する。ｒ＝２，３のｖｅｃｔｏｒ［ｒ］［０］［０：１］は以下のように計算される。

（ａ）ｖｅｃｔｏｒ［ｒ］［０］［０］＝（ｖｅｃｔｏｒ［０］［０］［０］×ｍ［ｐａｒｉｔｙ_ｒｅｆ］［ｐａｒｉｔｙ_ｐｒｅｄ］／／２）＋ｄｍｖｅｃｔｏｒ［０］
（ｂ）ｖｅｃｔｏｒ［ｒ］［０］［１］＝（ｖｅｃｔｏｒ［０］［０］［１］×ｍ［ｐａｒｉｔｙ_ｒｅｆ］［ｐａｒｉｔｙ_ｐｒｅｄ］／／２）＋ｅ［ｐａｒｉｔｙ_ｒｅｆ］［ｐａｒｉｔｙ_ｐｒｅｄ］＋ｄｍｖｅｃｔｏｒ［１］

【００８３】
ｍ［ｐａｒｉｔｙ_ｒｅｆ］［ｐａｒｉｔｙｐｒｅｄ］およびｅ［ｐａｒｉｔｙ_ｒｅｆ］［ｐａｒｉｔｙｐｒｅｄ］は，ＭＰＥＧ－２仕様（ＩＳＯ／ＩＥＣ１３８１８－２）においてそれぞれ，テーブル７－１１および７－１２に定義されている。
【００８４】
さらに，［ｒ］は，ＭＶのオーダ，たとえば，第一，第二といったオーダを示す。ｒ＝０はＭＶの第一のセットを示し，ｒ＝１はＭＶの第二のセットを示す。二重の主予想は，ＭＶの二つの余分なセットを特定するために，ｒ＝２およびｒ＝３を使用する。
【００８５】
“／／”は最も近い整数に繰り上げた，整数の区分を示す。
【００８６】
４．４　Ｂ－ＶＯＰの内ＭＶのコード化
付加的な変化が，ＭＰＥＧ－２ビットストリームのＢフレームにおける内ＭＶをコード化するとき（図４に示されているように）必要となる。ＭＰＥＧ－４は内モードをＢ－ＶＯＰに対する直接モードで置き換え，したがってＢフレームの内ＭＢがＭＰＥＧ－４シンタ

ックスにおいて，異なってコード化されなければならない。この問題には二つの解決策がある。

【００８７】

第一の解決策は，図３（全体の参照フレームに対してバッファがない）の前後トランスコーダに似たアーキテクチャーを用いることである。ＭＣは，ＭＢがその参照ＭＢ（その補償のないバージョン）に十分に近いことを前提として，同じＶＯＰの前のＭＢ（または，一つのＭＢのサイズをもつ余分なメモリーを費やしてテクスチャー残差を補償することなく前のＭＢ）に対して実行される。内
ＭＢに対するＭＶはＭＢの距離だけずれた前のＭＢのＭＶに等しい。

【００８８】

第二の解決策は，図４に示されたものに似たアーキテクチャーを使用することである。それはすべてのＩおよびＰ－ＶＯＰに対して参照フレームを保持する。ＭＣはこの解決策では，すべてのＰ－ＶＯＰで実行されなければならないことに留意されたい。内ＭＢに対するＭＶは予想されたＭＶ（三つの近傍にあるものの中央値）と同じであり，ＭＣは導出されたＭＶにより指定された参照ＭＢに対して実行される。

【００８９】

５．　圧縮ドメインにおけビデオのダウンスケール化

一般に，ビデオのダウンスケール化とサイズのトランスコード化は同じ意味をもつ。ダウンサンプル化は反エーリアシング（ローパス）フィルタでサブサンプル化を行うことを意味するが，サブサンプル化およびダウンサンプル化はここで，言い換え可能に使用されている。

【００９０】

サイズのトランスコード化は，その入力および出力が圧縮されたドメインにあるときに，コンピュータの使用で強力なものとなる。圧縮されたドメインで操作を制限する（実際上，デコード化およびエンコード化プロセスを避ける）ビデオダウンスケール化プロセスは複雑さを非常に減ずる。しかし，二つの新しい問題が，圧縮されたドメインにおけるダウンスケール化，すなわちＤＣＴ係数およびＭＶデータのダウンサンプル化とともに生じる。

【００９１】

最近，圧縮されたドメインでの，ビデオダウンスケール化アルゴリズムが議論されているが，ＭＰＥＧ－２とＭＰＥＧ－４の間の完全なトランスコード化（フィールド－フレームでインターレース化を含む）に関するものではない。本発明はこの問題を解決する。

【００９２】

サブセクション５．１および５．２はダウンサンプル化プロセスの二つの新しい問題への解決策を与える。本発明にしたがって，提案されたサイズトランスコーダの実施はセクション６ならびに図５および６で記述される。

【００９３】

５．１　ＤＣＴブロックのサブサンプル化

フレームに基づくビデオダウンスケール化において，四つの８×８ＤＣＴブロックを新しい８×８ＤＣＴブロック（フィールドブロックに関する詳細は下述する）に融合することが必要である。さらに，出力ブロックは入力ブロックのローパスバージョンであるべきである。このプロセスは，入力マトリクスにサブサンプル化マトリクス（好適にはローパスフィルタ）を掛けることにより，空間ドメインで実行される。空間ドメインでのサブサンプル化マトリクスによるかけ算は，直交変換の分配特性のため，ＤＣＴドメインでのサンプル化マトリクスのＤＣＴ係数によるかけ算に等しい。しかし，ダウンサンプル化フィルタに対する，ＤＣＴドメインにおけるダウンサンプル化プロセスの操作（計算）の数は，空間ドメインにおける対応する操作の総数と同じに多い。この問題の解決策は，疎であるダウンサンプル化（たてえば，比較的わずかなゼロでない値をもつマトリクス，たとえば３０パーセントかそれ以下）を用いることである。

【００９４】

疎ダウンサンプル化マトリクスがＤＣＴベースのベクトルとＤＣＴベースのベクトルの対称構造との間の直交特性に基づいてもよい。“Ａ Ｆａｓｔ Ｓｃｈｅｍｅ Ｆｏｒ Ｄｏｗｎｓａｍｐｌｉｎｇ Ａｎｄ Ｕｐｓａｍｐｌｉｎｇ Ｉｎ Ｔｈｅ ＤＣＴ Ｄｏｍａｉｎ”（Ｉｎｔｅｒｎａｔｉｏｎａｌ Ｃｏｎｆｅｒｅｎｃｅ ｏｎ Ｉｍａｇｅ Ｐｒｏｃｅｓｓｉｎｇ（ＩＣＩＰ）９９（ここに参考文献として組み込まれる））と題してＲ．ＤｕｇａｄおよびＮ．Ａｈｕｊａにより議論された一つのアプローチは，より低い４×４ＤＣＴ係数を四つの処理ブロックから取り，４×４ＩＤＣＴを各ＤＣＴサブブロックに適用し，新しい８×８画素ブロックを形成し，出力ブロックを得るために８×８ＤＣＴを適用する。ダウンサンプル化マトリクスは，ダウンサンプル化プロセスが固定されていることか，予め計算をすることができる。８×８ＤＣＴマトリクスを左半分と右半分に分けることにより，ダウンサンプル化マトリクス値の約半分が，４×４ＩＤＣＴマトリクスのカラムと左右両方の８×４ＤＣＴマトリクスのローとの間の直交性により，ゼロである。この操作（一次元）は数学的に以下のように書くことができる。

【数１】

$$B = Tb = T\begin{bmatrix} b_1 \\ \cdots \\ b_2 \end{bmatrix} = \begin{bmatrix} T_L & \vdots & T_R \end{bmatrix}\begin{bmatrix} T_4^t B_1 \\ \cdots \\ T_4^t B_2 \end{bmatrix} = T_L T_4^t B_1 + T_R T_4^t B_2$$

ここでｂは，８×１空間入力ベクトルであり，Ｂはその対応する８×１ＤＣＴベクトルであり，b_1およびb_2はサブサンプル化された４×１ベクトルで，B_1およびB_2はより低い４×１ＤＣＴベクトルで，Ｔは８×８ＤＣＴ変換マトリクスで，T_4は４×４ＤＣＴ変換マトリクスで，T_LおよびT_RはＴの左右半分である。添え字“ｔ”はマトリクス転置を示す。Ｄｕｇａｄのアルゴリズムはまた，ダウンサンプル化プロセスの複雑さを減ずるために，ＤＣＴベースのベクトルの対称特性を用いる。$T_L T^t_4$および$T_R T^t_4$は，Ｔの奇ローが反対称的で，Ｔの偶ローが対称できであることから，強度（$T_L T^t_4 = (-)^{i+j} T_R T^t_4$（ｉ，ｊ），$0 \leq i \leq 7$，$0 \leq j \leq 3$ の点に関して同じである。“ｉ”はマトリクスのローインデックスで，“ｊ”はマトリクスのカラムインデックスである。$T_L T^t_4$および$T_R T^t_4$の両方は，同じ成分，すなわち対称部分Ｅ（ｉ＋ｊが偶数のインデックス）および反対称部分Ｏ（ｉ＋ｊが奇数のインデックス）（$T_L T^t_4 = E + O$および$T_R T^t_4 = E - O$）に基づいて計算することができる。このような準備は，ダウンサンプル化プロセスが次のようになされたとき２の因数で，かけ算の数を減ずる。

$B = T_L T^t_4 B_1 + T_R T^t_4 B_2 = (E+O) B_1 + (E-O) B_2 = E(B_1 + B_2) + O(B_1 - B_2)$

【００９５】

四つのフィールドブロックを一つのフィールドブロックに変換するＤｕｇａｄの方法の実行はシンプルでない。このシナリオ（一次元）でのダウンサンプル化プロセスの拡張は次のように書くことができる。

$B = T(S_T T^t_4 B_T + S_B T^t_4 B_B)$

ここで，B_TおよびB_Bはより低い４×１フィールドブロック，S_TおよびS_Bはその頂部S_Tおよび底部S_Bフィールドブロックにそれぞれ対応する８×４でインターレース化されたマトリクスのＤＣＴ値である。S_Tの成分は，（$j = 2i$，$0 \leq i \leq 3$）のとき，$S_T(i, j) = 1$で，他のときは$S_T(i, j) = 0$である。S_Bの成分は，（$j = 2i + 1$，$0 \leq i \leq 3$）のとき，$S_B(i, j) = 1$で，他のときは$S_B(i, j) = 0$である。

【００９６】

これは，本発明にしたがって，ダウンサンプル化およびデインターレース化のためのＤｕｇａｄのアルゴリズムの修正である。

【００９７】

ダウンスケール化およびでインターレース化プロセスの操作は，ＳおよびＴが互いに直交ではないことからより複雑で，ダウンサンプル化マトリクスは疎ではない。Ｃ．ＹｉｍおよびＭ．Ａ．Ｉｓｎａｒｄｉの“Ａｎ Ｅｆｆｉｃｉｅｎｔ Ｍｅｔｈｏｄ Ｆｏｒ ＤＣＴ－Ｄｏｍａｉｎ Ｉｍａｇｅ Ｒｅｓｉｚｉｎｇ Ｗｉｔｈ Ｍｉｘｅｄ Ｆｉｅｌｄ／Ｆｒａｍｅ－Ｍｏｄｅ Ｍａｃｒｏｂｌｏｃｋｓ”（ＩＥＥＥ Ｔｒａｎｓ．Ｃｉｒｃ．ａｎｄ Ｓｙｓｔ．Ｆｏｒ Ｖｉｄｅｏ Ｔｅｃｈｎｏｌ．第９巻，第６９６－７００頁，１９９９年８月（ここに参考文献として組み込まれる））はフィールドブロックのダウンサンプル化の有効な方法を提案している。ローパスフィルタが，ダウンサンプル化マトリクス（Ｓ＝０．５［I_8 I_8］）が疎であるように，デインターレース化マトリクスに統合される。

【００９８】

I_8は８×８単位マトリクスを示し，［I_8 I_8］は，二つの単位マトリクスの連結からなる１６×８マトリクスを示す。単位マトリクスはもちろん，対角線上では全て１であり，他はゼロとなるものである。

【００９９】

本方法は，四つの８×８ＩＤＣＴフィールドブロックで開始し，ダウンサンプル化マトリクスを適用し，出力ブロックを得るために，８×８ＤＣＴを実行する。８×８ＩＤＣＴは４×４ＩＤＣＴに代えて，この方法で使用されることに留意されたい。この操作は，数学的に（一次元で）次のように示すことができる。

【数２】

$$D = TS\begin{bmatrix} T^t & 0 \\ 0 & T^t \end{bmatrix} = \frac{1}{2}T\begin{bmatrix} I_8 & \vdots & I_8 \end{bmatrix}\begin{bmatrix} T^t & 0 \\ 0 & T^t \end{bmatrix} = \frac{1}{2}\begin{bmatrix} I_8 & \vdots & I_8 \end{bmatrix}$$

【０１００】

５．２　ＭＶデータのサブサンプル化

ＭＥは全ビデオエンコード化プロセスの障害である。したがって，ＭＥ（全てのＭＢが間モードでコード化されるとする）を実際に実行することなく，四つの元々のＭＢのＭＶを使用することにより，サイズが変更したＭＢのＭＶを推定することは望ましい。ＭＰＥＧ－２ビットストリームが想定されると，各ＭＢが一つの入力をもつことから（唯一のＭＰＥＧ－４ビットストリームが全てのブロックに対してＭＶをもつことができる），ＭＶデータのサブサンプル化は四つのＭＢのＭＶをとる。最も単純な解決策は，新しいＭＶを得るために四つのＭＶを一緒にして平均化することであるが，これら四つのＭＶが異なるとき，推定できなくなる。Ｂ．Ｓｈｅｎ，Ｉ．Ｋ．ＳｅｔｈｉおよびＢ．Ｖａｓｕｄｅｖによる“Ａｄａｐｔｉｖｅ Ｍｏｔｉｏｎ－Ｖｅｔｏｒ Ｒｅｓａｍｐｌｉｎｇ ｆｏｒ Ｃｏｍｐｒｅｓｓｅｄ Ｖｉｄｅｏ Ｄｏｗｎｓｃａｌｌｉｎｇ，”（ＩＥＥＥ Ｔｒａｎｓ．Ｃｉｒｃ．ａｎｄ Ｓｙｓｔ．Ｆｏｒ Ｖｉｄｅｏ Ｔｅｃｈｎｏｌ．Ｖｏｌ．９，ｐｐ．９２９－９３６，１９９９年９月）は，よりよい結果が，最も悪い予想ＭＶにより重みを与えることより得ることができることを示す。各ＭＶの照合精度はそのＭＢのゼロでないＡＣ係数の数により示される。Ｓｈｅｎ等の技術を使用して，ダウンスケール化されたＭＢに対する新しいＭＶは次のように計算することができる。

【数３】

$$MV' = \frac{1}{2}\frac{\sum_{i=1}^{4} MV_i A_i}{\sum_{i=1}^{4} A_i}$$

【０１０１】

Ｍ．Ｒ．Ｈａｓｈｅｍｉ，Ｌ．ＷｉｎｇｅｒおよびＳ．Ｐａｎｃｈａｎａｔｈａｎによる“Ｃｏｍｐｒｅｓｓｅｄ Ｄｏｍａｉｎ Ｍｏｔｉｏｎ Ｖｅｃｔｏｒ Ｒｅｓａｍｐｌｉｎｇ Ｆｏｒ Ｄｏｗｎｓｃａｌｉｎｇ Ｏｆ ＭＰＥＧ Ｖｉｄｅｏ”（ＩＣＩＰ ９９）は，サイズが変更されたＭＢのＭＶを推定するための非線形方法を提案している。Ｓｈｅｎ等のアルゴリズムと似て，Ｈａｓｈｅｍｉの技術は，処理ＭＶの空間アクティビティを使用する。最大平均相関（ＭＡＣ）といわれている発見的測定が，出力ＭＶであるべき四つの元々のＭＶの一つを識別するためにＨａｓｈｅｍｉの方法において，用いられる。ＭＡＣを使用することにより，ダウンスケール化されたＭＢの新しいＭＶは次のように計算することができる。

【数４】

$$MV = \max \sum_{i=1}^{4} A_i \rho^{d_i}$$

ここでρは空間相関であり，０．８５とセットされ，d_iはＩ番目の入力ＭＶ（MV_i）と主力ＭＶとの間のユークリッド距離である。

【０１０２】

６．　サイズトランスコーダの実行

図５は，本発明にしたがったサイズトランスコーダを示す。Ｂフレームは入力ビットストリームに存在してもよいが，トランスコーダにより廃棄され，したがって出力には現れない。

【０１０３】

トランスコーダ５００において，ＭＶスケール化機能５１０，ＤＣＴスケール化機能５２０，空間スケール化機能５４０は追加されている。スイッチ５３０および５３５は，最初のセッティングに際して，ＤＣＴ機能４５５の出力が量子化機能３４０に至るようにされ，スイッチ５３５が閉じて，空間スケール化機能５４０の出力が加算器４４５へ入力されるように，調整される。スイッチ５３０および５３５の第二のセッティングに際し，ＤＣＴスケール化機能５２０の出力が量子化機能３４０に至り，スイッチ５３５が開く。

【０１０４】

トランスコーダ５００は，ＭＰＥＧ－２ビットストリームをより小さなサイズのビデオに対応するＭＰＥＧ－４ビットストリームに，たとえば，ＩＴＵ－Ｒ６０１からＳＩＦ（３５２×２４０）に変換する。

【０１０５】

ＭＰＥＧ－４に対する帯域の条件を達成するために，トランスコーダ５００はビデオを，（空間スケール化機能５４０で）水平方向および垂直方向の両方向に，二つサブサンプル化し，（時間スケール化機能５４５および５４６で）すべてのＢフレームをスキップし，これにより時間解像度を低下させる。時間スケール化機能５４６は代りとして，ＤＣＴスケール化機能５２０の後に，与えられていることに留意されたい。ダウンスケール化の実行の前に，Ｂフレームのスキップは複雑を軽減する。

【０１０６】

さらに，サブサンプル化の前の，ローパスフィルタ（時間スケール化機能５４０に設けることができる）により，画像の品質が改善される。

【０１０７】

本発明は，多少の修正をして，他のダウンサンプル化因子，およびＢ－ＶＯＰを含むように拡張できる。特に，ＭＶダウンスケール化およびモード決定に変更が加えられる。Ｂ－ＶＯＰに対するＭＶダウンスケール化は，後方ＭＶを含むために議論されたことの直接の拡張である。Ｂ－ＶＯＰに対するモード決定は，Ｐ－ＶＯＰでのと同じようにして（たとえば，内ＭＢをＰ－ＶＯＰの間ＭＢに変換するように，単一の方向のＭＶを二方向のＭＶに変換することにより），取り扱うことができる。

【０１０８】

以下で，サイズトランスコーダ５００に関連した六つの問題が議論される。また，入力ビデオは７０４×４８０画素解像度で，ＭＰ@ＭＬ ＭＰＥＧ－２エンコーダでコード化され，所望の出力は，ＳＩＦ順次ビデオ（Ｎだけフレームレートの減少をともなう）を含む，単純なプロファイルのＭＰＥＧ－４ビットストリームであるとしている。

【０１０９】

６．１ ビデオＭＶダウンスケール化（ルミナンス）

この問題は，四つすべてのＭＢが相互のように，コード化され，フレーム予想を使用するときに現れる。これらＭＢの各ＭＶは，ＭＰＥＧ－４（ＭＰＥＧ－４が８×８ブロック当たり一つのＭＶを可能にする）の四つのブロックのＭＶを決定するために，各方向（水平および垂直）で二の率でダウンスケール化される。スケールが決められたＭＶはつぎに，通常のＭＰＥＧ－４の手順を使用して（中央値フィルタを使用して）予想してエンコード化される。

【０１１０】

各ＭＢ（四つのブロックからなる）はＭＰＥＧ－２とＭＰＥＧ－４の両方において，同じモードでコード化されなければならない。ビデオのダウンスケール化でもって，出力ＭＢ（四つのブロック）は四つの入力ＭＢに対応する。

【０１１１】

６．２ インターレース化されたビデオＭＶダウンサンプル化（輝度）

この問題は，四つすべてのＭＢが間のようにコード化され，フィールド予想を使用するときに存在する。サイズが変更されたブロックのフレームＭＶを得るために，各ＭＢの二つのフレームＭＶを組み合わせる必要がある。空間アクティビティに基づいて新しいＭＶをセットすることに代えて，提案のトランスコーダは，近傍のＭＢに基づいて新しいＭＶを取る。取り巻く八つ全てのＭＢのＭＶは予想子（フィールドＭＶがフィールド予想を有するＭＢの場合に平均化される）を見いだすために使用される。これら八つのＭＶからの中央値は予想子となり，現在のＭＢ（ユークリッド距離に関してより接近した）のフィールドＭＶは，新しいＭＶとなるように水平方向で，二の率でサイズの変更がなされる。

【０１１２】

６．３ ＭＶダウンサンプル化（クロミナンス）

この問題は，四つすべてのＭＢが間としてコード化され，フレームまたはフィールド予想（ＭＰＥＧ－４はクロミナンスブロックに対して同じ方法で，両予想モードを処理する）のいずれかを使用したときに生ずる。プロセスは，ルミナンスＭＶからクロミナンスＭＶ（すなわち，クロミナンスＭＶがその四つの対応する８×８ルミナンスＭＶの平均の，ダウンスケール化されたバージョンである）を得るために，ＭＰＥＧ－４の方法に従う。

【０１１３】

６．４ ＤＣＴダウンサンプル化（ルミナンスプログレッシブ，クロミナンス）

この問題は，四つすべてのルミナンスＭＢが内または間としてコード化され，フレームＭＢ構造を使用するときに生じ，それら八つのクロミナンスブロック（Ｃｒに対して四つ，Ｃｂに対して四つ）はフレームかフィールド構想のいずれかを使用する。Ｄｕｇａｄの方法は，各方向に，２の率で，ルミナンスおよびクロミナンスＤＣＴブロックをダウンスケール化するために使用される。

【０１１４】

６．５ インターレース化されたＤＣＴダウンサンプル化（ルミナンス）

この問題は，二つの方法の一方で現れる。第一に，その関連したＭＢはフィールド予想を使用し，第二に，その関連したＭＢはフレーム予想を使用する。いずれの場合も，四つの８×８フィードＤＣＴブロック（頂部フィールドに対して二つ，底部フィールドに対して二つ）を，一つのフレームＤＣＴブロックにダウンスケール化することが望まれる。第一の場合の解決策は，ＭＣに対して選択されたものと同じフィールドＤＣＴブロックを使用することである。第二の場合はデインターレース化に係り，上述した，ＤｕｇａｄおよびＹｉｍの方法の組み合わせが提案される。

【０１１５】

特に，トランスコーダはまず，二つのフレームブロックを得るために，Ｙｉｍアルゴリズムに基づいて，垂直方法に四つのフィールドブロックをダウンスケール化する（同時にデインターレース化を実行する）。トランスコーダはつぎに，Ｄｕｇａｒのアルゴリズムを使用して，出力ブロックを得るために，水平方向に，これら二つのフレームブロックをダウンスケール化する。

【０１１６】

このことは，図６に示されているが，ここで四つの８×８係数フィールドモードＤＣＴブロックは符号６００で示され，二つの８×８フレームモードＤＣＴブロックは符号６１０で示され，一つの８×８フレームモードＤＣＴブロックは符号６２０で示されている。

【０１１７】

本発明にしたがった，ＤＣＴダウンスケール化のための手順は，以下の通りに要約できる。

１．　符号６００で示されているように，四つのフィールドブロックを一緒にして組み合わせることにより，１６×１６係数入力マトリクスを形成する。

２．　垂直ダウンスケール化およびフィルタリング化に対して，Ｙｉｍのアルゴリズムにしたがってローパス（ＬＰ）フィルタＤを入力マトリクスのロー毎に適用する。ＬＰ入力マトリクスは符号６１０で示されているように，今や１６×８画素である。

３．　次のＬＰマトリクスからB_1およびB_2　８×８マトリクスを形成する。

【数５】

$$\left(\left[B_1 \vdots B_2\right]\right).$$

４．　次のように，出力マトリクス（８×８）（６２０）を得るために，B_1およびB_2のカラム毎に，Ｄｕｇａｄのアルゴリズムにしたがて，水平ダウンスケール化操作を実行する。

$V = B_1 (T_L T^t_4)^t + B_2 (T_R T^t_4)^t = (B_1 + B_2) E + (B_1 - B_2) O$

ここで，ＥおよびＯは上述した偶数および奇数のローを示す。

【０１１８】

特に，以下のように奇“Ｏ”および偶“Ｅ”マトリクスからなる，水平ダウンサンプル化マトリクスが使用され得る（スケール化率は無視）。

Ｅ＝［ｅ（０）　０　０　０
０　ｅ（１）　０　ｅ（２）
０　０　０　０
０　ｅ（３）　０　ｅ（４）
０　０　ｅ（５）　０
０　ｅ（６）　０　ｅ（７）
０　０　０　０
０　ｅ（８）　０　ｅ（９）］

Ｏ＝［０　０　０　０
ｏ（０）　０　ｏ（１）　０
０　ｏ（２）　０　０
ｏ（３）　０　ｏ（４）　０
０　０　０　０
ｏ（５）　０　ｏ（６）　０
０　０　０　ｏ（７）
ｏ（８）　０　ｏ（９）　０］

次のように係数は使用できる。

ｅ（０）＝４　　ｏ（０）＝２．５６９１５４４８
ｅ（１）＝０．８３１４６９６１２　　ｏ（１）＝－０．１４９１５６６８

ｅ（２）＝　０．０４５７７４６５４　　ｏ（２）＝　２
ｅ（３）＝　１．５８２１３０１６７　　ｏ（３）＝－０．８９９９７６２２３
ｅ（４）＝－０．１９５０９０３２２　　ｏ（４）＝　１．０２６５５９９３４
ｅ（５）＝　２　　ｏ（５）＝　０．６０１３４４８８７
ｅ（６）＝－０．７０４８８５９０１　　ｏ（６）＝　１．５３６３５５１３
ｅ（７）＝　０．９８０７８５２８０　　ｏ（７）＝　２
ｅ（８）＝　０．９０６１２７４４６　　ｏ（８）＝－０．５０９７９５５７９
ｅ（９）＝　１．７３１４４５８３５　　ｏ（９）＝－０．７５０６６０５５５

【０１１９】

要するに，疎であるＤＣＴマトリクスの積は，ダウンサンプル化するマトリクスとして使用される。

【０１２０】

この技術は，四つのＮ／２×Ｎ／２係数フィールドブロックからなる，Ｎ×Ｎブロックの２：１のダウンサイズ化のために，一般的に拡張できる。他のダウンサイズ化の比も順応できる。

【０１２１】

６．６　特別な場合

特別な場合は，四つ全てのＭＶが同じモード（前の五つの場合のいずれにも収まらない）でコード化されない時に生ずる。どの内または他の間ＭＢの中のスキップしたＭＢは間モードであると，常に想定されている。フィールドＭＶはフレームＭＶを得るために，セクション６．２に基づいて融合され，つぎに，セクション６．１の技術が適用される。ＭＣは内ブロックのテクスチャー（トランスコーダにより，ゼロＭＶをもつ相互ブロックとして処理される）を決定するために，推奨される。

【０１２２】

７．　まとめ

本発明が，エラーが些細で，複雑性が可能な限り低いトランスコーダアーキテクチャーを提供することは分かるであろう。このエラーは，ＭＰＥＧ－４テクスチャーエンコード化プロセス（ＱＰコード化，ＤＣ予想，非線形ＤＣスケーラ）において生ずる。これらプロセスは，ほとんど損失のないトランスコード化システムを形成するために，ＭＰＥＧ－４の今後のプロファイルで，除去されるべきである。

【０１２３】

本発明はまた，ＭＰＥＧ－２ ＭＰ@ ＭＬでコード化する，ＩＴＵ－Ｒ６０１インターレース化されたビデオを，ストリーミングビデオの場合に対して適したＳＩＦプログレッシブビデオを含む単純なプロファイルＭＰＥＧ－４ビットストリームに変換するために，サイズトランスコーダの完全な詳細を提供する。

【０１２４】

フィールドモードのＤＣＴブロックの空間ダウンスケール化に対して，疎のダウンサンプル化マトリクスが垂直および水平方向に使用され，これによりトランスコーダの計算が減少する新規な方法で，垂直および水平ダウンスケール化技術を組み合わせることが提案された。

【０１２５】

さらに，ＭＶダウンスケール化に対して，八つの近傍のＭＶからの中央値を使用することが提案された。この提案は，本件予想されたＭＶがグローバルＭＶと同調することから，セクション５．２のアルゴリズムよりよりよく働く。

【０１２６】

本発明は，好適実施例に関連して記述されてきたが，種々の修正，適応例が本特許請求の範囲の発明の範囲から逸脱することなくなし得ることは分かるであろう。

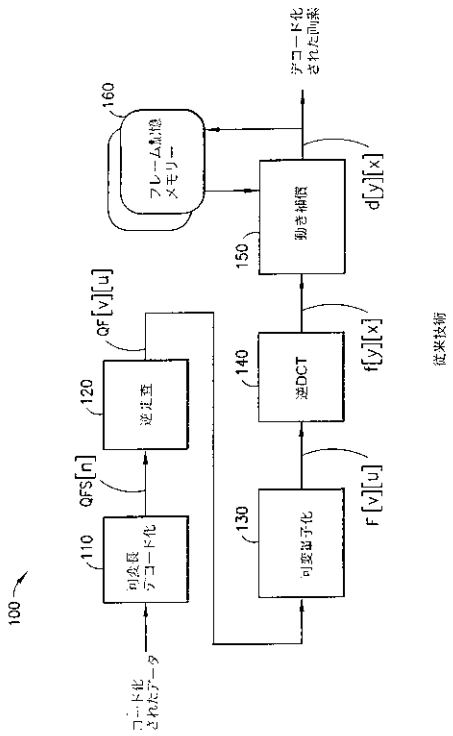
【図面の簡単な説明】

【図１】

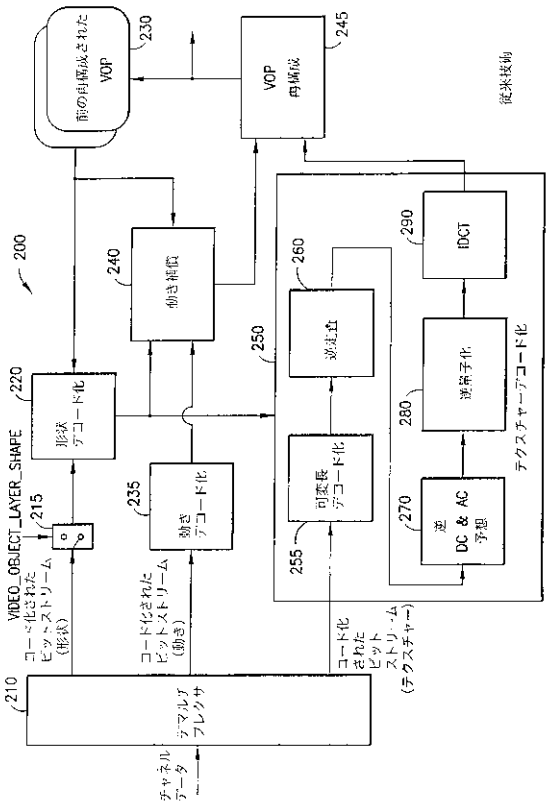
図１は，ＭＰＥＧ－２ビデオデコーダを示す。

【図２】
図２は，拡張性の特徴をもたないＭＰＥＧ－４ビデオデコーダを示す。
【図３】
図３は，本発明にしたがった，複雑さが低い，前後トランスコーダ（使用禁止になったＢフレームをもつ）を示す。
【図４Ａ】
図４（Ａ）は，本発明にしたがって，ドリフトエラーを最小にした（使用可能なＢフレームをもつ）トランスコーダアーキテクチャーを示す。
【図４Ｂ】
図４（Ｂ）は，本発明にしたがって，ドリフトエラーを最小にした（使用可能なＢフレームをもつ）トランスコーダアーキテクチャーを示す。
【図５Ａ】
図５（Ａ）は，本発明にしたがった，サイズトランスコーダを示す。
【図５Ｂ】
図５Ｂは，本発明にしたがった，サイズトランスコーダを示す。
【図６】
図６は，本発明にしたがった，四つのフィールドモードＤＣＴブロックを一つのフレームモードＤＣＴブロックにダウンサンプル化することを示す。

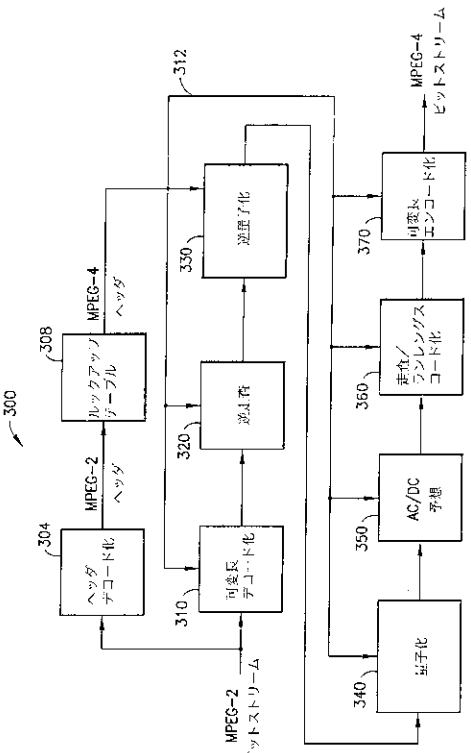
【図１】



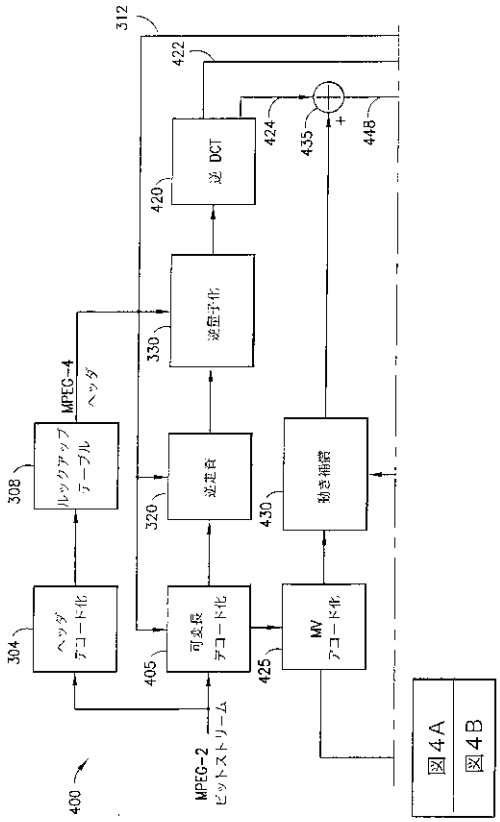
【図２】



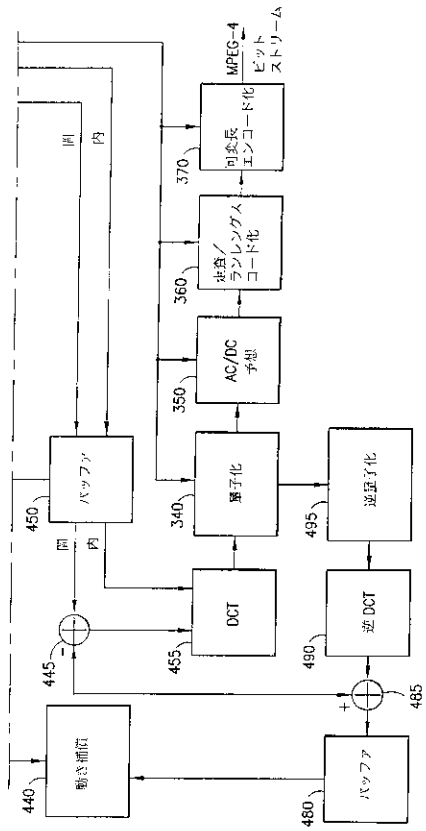
【図３】



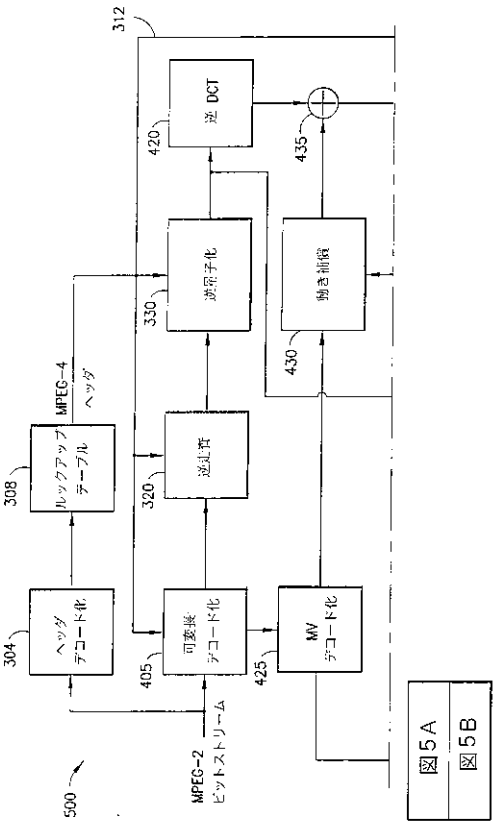
【図４Ａ】



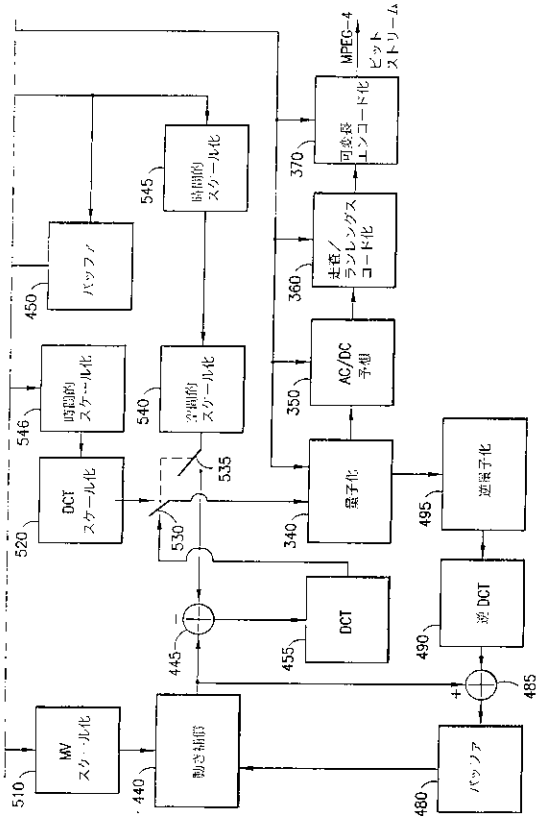
【図４Ｂ】



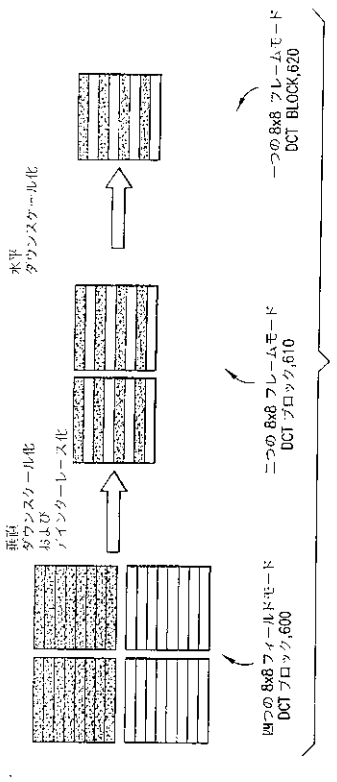
【図５Ａ】



【図５Ｂ】


MPEG-4
ビット
ストリーム
可変長
エンコード化
370
走査/
ランレングス
コード化
360
AC/DC
予想
350
量子化
340
逆量子化
495
逆 DCT
490
485
パッファ
480
動き補償
440
MV
スケール化
510
DCT
スケール化
520
時間的
スケール化
546
パッファ
450
時間的
スケール化
545
空間的
スケール化
540
535
530
DCT
455
445

【図６】


一つの 8x8 フレームモード
DCT BLOCK,620
水平
ダウンスケール化
二つの 8x8 フレームモード
DCT ブロック,610
垂直
ダウンスケール化
および
ノインターレース化
四つの 8x8 フィールドモード
DCT ブロック,600

【国際公開パンフレット】

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
13 December 2001 (13.12.2001)

PCT

(10) International Publication Number
WO 01/95633 A2

(51) International Patent Classification⁷: H04N 7/26

(21) International Application Number: PCT/US01/10811

(22) International Filing Date: 29 May 2001 (29.05.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
09/590,928 9 June 2000 (09.06.2000) US

(71) Applicant (*for all designated States except US*): GENERAL INSTRUMENT CORPORATION [US/US]; 101 Tournament Drive, Horsham, PA 19044 (US).

(72) Inventors; and

(75) Inventors/Applicants (*for US only*): PANUSOPONE, Krit [TH/US]; 9656 Carroll Canyon Road #F1, San Diego, CA 92126 (US). CHEN, Xuemin [US/US]; 8560 Foxcroft Place, San Diego, CA 92129 (US).

(74) Agent: LIPSITZ, Barry, R.; Law Offices of Barry R Lipsitz, Building No. 8, 755 Main Street, Monroe, CT 06468 (US).

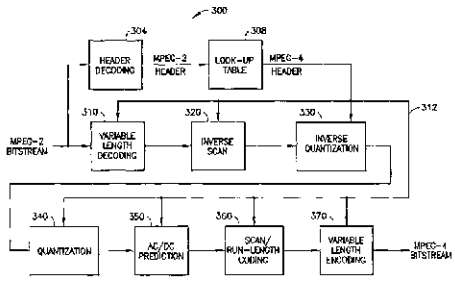
(81) Designated States (*national*): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States (*regional*): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW). Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM). European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR). OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:
without international search report and to be republished upon receipt of that report

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: VIDEO SIZE CONVERSION AND TRANSCODING FROM MPEG-2 TO MPEG-4



(57) Abstract: A transcoder architecture that provides the lowest possible complexity with a small error, e.g., for converting an MPEG-2 bitstream into an MPEG-4 bitstream. The transcoder reads header information (304) from an input bitstream and provides a corresponding header in the new format for the output bitstream. In one embodiment (Fig. 3), a low complexity front-to-back transcoder (with B frames disabled) avoids the need for motion compensation processing. In another embodiment (Fig. 4), a transcoder architecture that minimizes drift error (with B frames enabled) is provided. In another embodiment (Fig. 5), a size transcoder (with B frames enabled) is provided, e.g., to convert a bitstream of ITU-R 601 interlaced video coding with MPEG-2 MP@ ML into a simple profile MPEG-4 bitstream which contains SIF progressive video suitable for a streaming video application. For spatial downscaling of field mode DCT blocks, vertical and horizontal downscalling techniques are combined to use sparse matrixes to reduce computations.

WO 01/95633 A2

VIDEO SIZE CONVERSION AND TRANSCODING FROM MPEG-2 TO MPEG-4

BACKGROUND OF THE INVENTION

The present invention relates to compression of multimedia data and, in particular, to a video transcoder that allows a generic MPEG-4 decoder to decode MPEG-2 bitstreams. Temporal and spatial size conversion (downscaling) are also provided.

The following acronyms and terms are used:

CBP - Coded Block Pattern
DCT - Discrete Cosine Transform
DTV - Digital Television
DVD - Digital Video Disc
HDTV - High Definition Television
FLC - Fixed Length Coding
IP - Internet Protocol
MB - Macroblock
ME - Motion Estimation
ML - Main Level
MP - Main Profile
MPS - MPEG-2 Program Stream
MTS - MPEG-2 Transport Stream
MV - Motion Vector
QP - quantization parameter
PMV - Prediction Motion Vector
RTP - Real-Time Transport Protocol (RFC 1889)
SDTV - Standard Definition Television

SIF - Standard Intermediate Format
SVCD - Super Video Compact Disc
VLC - Variable Length Coding
VLD - Variable Length Decoding
VOP - Video Object Plane

MPEG-4, the multimedia coding standard, provides a rich functionality to support various applications, including Internet applications such as streaming media, advertising, interactive gaming, virtual traveling, etc. Streaming video over the Internet (multicast), which is expected to be among the most popular application for the Internet, is also well-suited for use with the MPEG-4 visual standard (ISO/IEC 14496-2 Final Draft of International Standard (MPEG-4), "Information Technology - Generic coding of audio-visual objects, Part 2: visual," Dec. 1998).

MPEG-4 visual handles both synthetic and natural video, and accommodates several visual object types, such as video, face, and mesh objects. MPEG-4 visual also allows coding of an arbitrarily shaped object so that multiple objects can be shown or manipulated in a scene as desired by a user. Moreover, MPEG-4 visual is very flexible in terms of coding and display configurations by including enhanced features such as multiple auxiliary (alpha) planes, variable frame rate, and geometrical transformations (sprites).

However, the majority of the video material (e.g., movies, sporting vents, concerts, and the like) which is expected to be the target of streaming video is

already compressed by the MPEG-2 system and stored on storage media such as DVDs, computer memories (e.g., server hard disks), and the like. The MPEG-2 System specification (ISO/IEC 13818-2 International Standard (MPEG-2), "Information Technology - Generic coding of Moving Pictures and Associated Audio: Part 2 - Video," 1995) defines two system stream formats: the MPEG-2 Transport Stream (MTS) and the MPEG-2 Program Stream (MPS). The MTS is tailored for communicating or storing one or more programs of MPEG-2 compressed data and also other data in relatively error-prone environments. One typical application of MTS is DTV. The MPS is tailored for relatively error-free environments. The popular applications include DVD and SVCD.

Attempts to address this issue have been unsatisfactory to date. For example, the MPEG-4 studio profile (O. Sunohara and Y. Yagasaki, "The draft of MPEG-4 Studio Profile Amendment Working Draft 2.0," ISO/IEC JTC1/SC29/WG11 MPEG99/5135, Oct. 1999) has proposed a MPEG-2 to MPEG-4 transcoder, but that process is not applicable to the other MPEG-4 version 1 profiles, which include the Natural Visual profiles (Simple, Simple Scaleable, Core, Main, N-Bit), Synthetic Visual profiles (Scaleable Texture, Simple Face Animation), and Synthetic/Natural Hybrid Visual (Hybrid, Basic Animated Texture). The studio profile is not applicable to the Main Profile of MPEG-4 version 1 since it modifies the syntax, and the decoder process

is incompatible with the rest of the MPEG-4 version 1 profiles.

The MPEG standards designate several sets of constrained parameters using a two-dimensional ranking order. One of the dimensions, called the "profile" series, specifies the coding features supported. The other dimension, called "level", specifies the picture resolutions, bit rates, and so forth, that can be accommodated.

For MPEG-2, the Main Profile at Main Level, or MP@ML, supports a 4:2:0 color subsampling ratio, and I, P and B pictures. The Simple Profile is similar to the Main Profile but has no B-pictures. The Main Level is defined for ITU-R 601 video, while the Simple Level is defined for SIF video.

Similarly, for MPEG-4, the Simple Profile contains SIF progressive video (and has no B-VOPs or interlaced video). The Main Profile allows B-VOPs and interlaced video.

Accordingly, it would be desirable to achieve interoperability among different types of end-systems by the use of MPEG-2 video to MPEG-4 video transcoding and/or MPEG-4-video to MPEG-2-video transcoding. The different types of end-systems that should be accommodated include:

Transmitting Interworking Unit (TIU): Receives MPEG-2 video from a native MTS (or MPS) system and transcodes to MPEG-4 video and distributes over packet networks using a native RTP-based system layer (such as

an IP-based internetwork). Examples include a real-time encoder, a MTS satellite link to Internet, and a video server with MPS-encoded source material.

Receiving Interworking Unit (RIU): Receives MPEG-4 video in real time from an RTP-based network and then transcodes to MPEG-2 video (if possible) and forwards to a native MTS (or MPS) environment. Examples include an Internet-based video server to MTS-based cable distribution plant.

Transmitting Internet End-System (TIES): Transmits MPEG-2 or MPEG-4 video generated or stored within the Internet end-system itself, or received from internet-based computer networks. Examples include a video server.

Receiving Internet End-System (RIES): Receives MPEG-2 or MPEG-4 video over an RTP-based internet for consumption at the Internet end-system or forwarding to a traditional computer network. Examples include a desktop PC or workstation viewing a training video.

It would be desirable to determine similarities and differences between MPEG-2 and MPEG-4 systems, and provide transcoder architectures which yield a low complexity and small error.

The transcoder architectures should be provided for systems where B-frames are enabled (e.g., main profile), as well as a simplified architecture for when B-frames are not used (simple profile).

Format (MPEG-2 to MPEG-4) and/or size transcoding should be provided.

It would also be desirable to provide an efficient mapping from the MPEG-2 to MPEG-4 syntax, including a mapping of headers.

The system should include size transcoding, including spatial and temporal transcoding.

The system should allow size conversion at the input bitstream or output bitstream of a transcoder.

The size transcoder should convert a bitstream of ITU-R 601 interlaced video coded with MPEG-2 MP@ML into a simple profile MPEG-4 bitstream which contains SIF progressive video suitable, e.g., for a streaming video application.

The system should provide an output bitstream that can fit in the practical bandwidth for a streaming video application (e.g., less than 1 Mbps).

The present invention provides a system having the above and other advantages.

SUMMARY OF THE INVENTION

The invention relates to format transcoding (MPEG-2 to MPEG-4) and size (spatial and temporal) transcoding.

A proposed transcoder includes size conversion, although these parameters can be transcoded either at the input bitstream or the output bitstream. However, it is more efficient to include all kinds of transcoding into the product version of a transcoder to reduce the complexity since the transcoders share processing elements with each other (such as a bitstream reader).

The invention addresses the most important requirements for a transcoder, e.g., the complexity of the system and the loss generated by the process.

In one embodiment, a proposed front-to-back transcoder architecture reduces complexity because there is no need to perform motion compensation.

In a particular embodiment, the transcoder can use variable 5-bit QP representation, and eliminates AC/DC prediction and the nonlinear DC scaler.

The invention is alternatively useful for rate control and resizing.

A particular method for transcoding a pre-compressed input bitstream that is provided in a first video coding format includes the steps of: recovering header information of the input bitstream; providing corresponding header information in a second, different

video coding format; partially decompressing the input bitstream to provide partially decompressed data; and re-compressing the partially decompressed data in accordance with the header information in the second format to provide the output bitstream.

A method for performing 2:1 downscaling on video data includes the steps of: forming at least one input matrix of NxN (e.g., N=16) Discrete Cosine Transform (DCT) coefficients from the video data by combining four N/2xN/2 field-mode DCT blocks; performing vertical downsampling and de-interlacing to the input matrix to obtain two N/2xN/2 frame-mode DCT blocks; forming an NxN/2 input matrix from the two frame-mode DCT blocks; and performing horizontal downsampling to the NxN/2 matrix to obtain one N/2xN/2 frame-mode DCT block.

Preferably, the vertical and horizontal downsampling use respective sparse downsampling matrixes. In particular, a vertical downsampling matrix of $0.5[I_8\ I_8]$ may be used, where I_8 is an 8x8 identity matrix. This is essentially vertical pixel averaging. A horizontal downsampling matrix composed of odd "O" and even "E" matrices may be used.

Corresponding apparatuses are also presented.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 illustrates an MPEG-2 video decoder.

FIG. 2 illustrates an MPEG-4 video decoder without any scalability feature.

FIG. 3 illustrates a low complexity front-to-back transcoder (with B frames disabled) in accordance with the invention.

FIG. 4 illustrates a transcoder architecture that minimizes drift error (with B frames enabled) in accordance with the invention.

FIG. 5 illustrates a size transcoder in accordance with the invention.

FIG. 6 illustrates downsampling of four field mode DCT blocks to one frame mode DCT block in accordance with the present invention.

DETAILED DESCRIPTION OF THE INVENTION

The invention relates to format transcoding (MPEG-2 to MPEG-4) and size (spatial and temporal) transcoding.

The invention provides bit rate transcoding to convert a pre-compressed bitstream into another compressed bitstream at a different bit rate. Bit rate transcoding is important, e.g., for streaming video applications because the network bandwidth is not constant and, sometimes, a video server needs to reduce the bit rate to cope with the network traffic demand. A cascaded-based transcoder which re-uses MVs from the input bitstream and, hence, eliminates motion estimation (ME), is among the most efficient of the bit rate transcoders. The cascaded-based transcoder decodes the input bitstream to obtain the MV and form the reference frame. It then encodes this information with a rate control mechanism to generate an output bitstream at the desired bit rate.

Spatial resolution transcoding becomes a big issue with the co-existence of HDTV and SDTV in the near future. It is also very beneficial for the streaming video application since it is likely that the Internet bandwidth is not going to be large enough for broadcast quality video. Hence, downsampling of the broadcast quality bitstream into a bitstream with a manageable resolution is appealing. Spatial resolution transcoding usually performs in the compressed (DCT)

domain since it drastically reduces the complexity of the system. The process of downsampling in the compressed domain involves the processing of two parameters, namely DCT coefficients and MVs. A downsampling filter and its fast algorithm is suggested to perform DCT coefficient downsampling. MV resampling is used to find the MV of the downsampled video. In the real product, to avoid drift, the residual of the motion compensation should be re-transformed instead of approximating the DCT coefficients from the input bitstream.

2. High level comparison

Structure-wise, MPEG-2 and MPEG-4 employ a similar video compression algorithm. Fundamentally, both standards adopt motion prediction to exploit temporal correlation and quantization in the DCT domain to use spatial correlation within a frame. This section describes the structure of the MPEG-2 and MPEG-4 decoders at a high level, and then notes differences between the two standards.

2.1 MPEG-2

FIG. 1 shows the simplified video decoding process of MPEG-2. In the decoder 100, coded video data is provided to a variable length decoding function 110 to provide the one-dimensional data QFS[n], where n is a coefficient index in the range of 0-63. At the inverse scan function 120, QFS[n] is converted into a two-dimensional array of coefficients denoted by QF[v][u], where the array indexes u and v both lie in the range 0

to 7. An inverse quantisation function 130 applies the appropriate inverse quantisation arithmetic to give the final reconstructed, frequency-domain DCT coefficients, F[v][u]. An inverse DCT function 140 produces the pixel (spatial) domain values f[y][x]. A motion compensation function 150 is responsive to a frame store memory 160 and the values f[y][x] for producing the decoded pixels (pels) d[y][x], where y and x are Cartesian coordinates in the pixel domain.

MPEG-2 operates on a macroblock level for motion compensation, a block level for the DCT transformation, and the coefficient level for run-length and lossless coding. Moreover, MPEG-2 allows three types of pictures, namely I-, P- and B- pictures. Allowed motion prediction modes (forward, backward, bi-directional) are specified for the P- and B- pictures. MPEG-2 uses interlaced coding tools to handle interlaced sources more efficiently.

2.2 MPEG-4

FIG. 2 shows the MPEG-4 video decoding process without any scalability features.

At the decoder 200, data from a channel is output from a demux 210. A coded bit stream of shape data is provided to a switch 215, along with the MPEG-4 term video_object_layer_shape (which indicates, e.g., whether or not the current image is rectangular, binary only, or grayscale). If video_object_layer_shape is equal to "00" then no binary shape decoding is

required. Otherwise, binary shape decoding is carried out.

If binary shape decoding is performed, a shape decoding function 220 receives the previous reconstructed VOP 230 (which may be stored in a memory), and provides a shape-decoded output to a motion compensation function 240. The motion compensation function 240 receives an output from a motion decoding function 235, which, in turn, receives a motion coded bit stream from the demux 210. The motion compensation function 240 also receives the previous reconstructed VOP 230 to provide an output to a VOP reconstruction function 245.

The VOP reconstruction function 245 also receives data from a texture decoding function 250 which, in turn, receives a texture coded bit stream from the demux 210, in addition to an output from the shape decoding function 220. The texture decoding function 250 includes a variable length decoding function 255, an inverse scan function 260, an inverse DC and AC prediction function 270, an inverse quantization function 280 and an Inverse DCT (IDCT) function 290.

Compared to MPEG-2, several new tools are adopted in MPEG-4 to add features and interactivity, e.g., sprite coding, shape coding, still texture coding, scalability, and error resilience. Moreover, motion compensation and texture coding tools in MPEG-4, which are similar to MPEG-2 video coding, are modified to improve the coding efficiency, e.g., coding tools such

as direct mode motion compensation, unrestricted motion compensation, and advanced prediction.

In particular, direct mode motion compensation is used for B-VOPs. Specifically, it uses direct bi-directional motion compensation derived by employing I- or P-VOP macroblock MVs and scaling them to derive forward and backward MVs for macroblocks in B-VOP. Only one delta MV is allowed per macroblock. The actual MV is calculated from the delta vector and the scaled MV from its co-located macroblock.

Unrestricted motion compensation allows one or four MVs per macroblock. The four MV mode is only possible in B-VOPs with the use of direct mode. Note that the MV for a chrominance macroblock is the average of four MVs from its associated luminance macroblock. Furthermore, unrestricted motion compensation allows an MV to point out of the reference frame (the out-of-bound texture is padded from the edge pixel).

Advanced prediction defines the prediction method for MV and DCT coefficients. A MV predictor is set according to the median value of its three neighbors' MVs. Prediction of the intra DCT coefficient follows the intra AC/DC prediction procedure (Graham's rule).

3. Transcoder architecture

FIG. 3 illustrates a low complexity front-to-back transcoder in accordance with the invention, with B frames disabled.

Similarities between the structures of MPEG-2 and MPEG-4 allow a low complexity (front-to-back)

transcoder. Instead of completely decoding an MPEG-2 bitstream to the spatial (pixel) domain level, the front-to-back transcoder 300 uses DCT coefficients and MVs to generate an MPEG-4 bitstream without actually performing a motion estimation process. A trade-off is that this architecture may cause a drift in the reconstructed frame, and does not allow bit rate control. However, the drift problem is small since most of the difference between the MPEG-2 and MPEG-4 decoders lies in the lossless coding part.

The transcoder 300 comprises a cascade of a MPEG-2 bitstream reader (decoder) (310-330) and a MPEG-4 header and texture coder (encoder) (340-370), along with a header decoding function 304, a look-up table 308, and a communication path 312. The transcoder 300 reads an input MPEG-2 bitstream, performs a variable length decoding (VLD) at a function 310 on DCT coefficients and MV residual, and then follows MPEG-2 logic to find DCT coefficients and/or MVs of every block in the frame.

The header decoding function 304 decodes the MEPG-2 headers and provides them to a look-up table (or analogous function) 308, which uses the tables detailed below to obtain corresponding MPEG-4 headers.

With the information of the headers, DCT coefficients and/or MV, the transcoder 300 encodes this information into the MPEG-4 format. Note that the reference frame is not needed in this architecture.

The transcoder 300 reads the MPEG-4 header from the input bitstream and writes the corresponding MPEG-4 header in its place in an output bitstream.

After processing at the VLD 310, the data is provided to an inverse scan function 320, and an inverse quantisation function 330. Next, using the MPEG-4 header information provided via the path 312, the decoded, DCT coefficient data is processed at a MPEG-4 header and texture coder that includes a quantisation function 340, and an AC/DC prediction function 350 for differentially encoding the quantised DCT coefficients. In particular, the AC/DC prediction process generates a residual of DC and AC DCT coefficients in an intra MB by subtracting the DC coefficient and either the first row or first column of the AC coefficients. The predictor is adaptively selected. Note that the AC/DC prediction function 350 may not need the MPEG-4 header information.

Subsequently, a scan/run-length coding function 360 and a variable length encoding function 370 provide the MPEG-4 bitstream.

FIG. 4 illustrates a transcoder architecture that minimizes drift error in accordance with the invention, with B frames enabled.

Like-numbered elements correspond to one another in the figures.

To counter the problems of drift in the reconstructed frame, and the lack of bit rate control, a more complex architecture such as the transcoder 400,

which is an extension of the transcoder 300 of FIG. 3, can be used. This architecture actually computes the DCT coefficient of the texture/residual data, hence motion compensation is required. Since the encoder of this transcoder includes a decoding process, the drift error can be minimized.

Moreover, the transcoder 400 can be used to transcode bitstreams with B-frames since MPEG-4 does not allow intra mode for B-frames. The transcoder 400 treats a block in intra mode in a B-frame (in MPEG-2) as a block with a zero MV in inter mode (in MPEG-4). It can be either a zero residual MV (PMV) or zero MV (which may yield a non-zero MV code) since the MV is predictive coded against the PMV.

In particular, the transcoder 400 includes a variable length decoding function 405 that provides MV residue data to a MV decoder 425, and that provides DCT coefficient data to the inverse scan function 320. The DCT data is processed by the inverse quantisation function 330 and an inverse DCT function 420 to obtain pixel domain data. Intra-coded pixel data is provided via a path 422 to a buffer, while inter-coded pixel data is provided to an adder 435 via a path 424.

The pixel (difference) data on path 424 is added to reference pixel data from a motion compensation function 430 (responsive to the MV decoder 425) to provide inter-coded data to the buffer 450 via a path 448.

For re-encoding, e.g., in the MPEG-4 format, the buffer 450 either outputs the intra pixel data directly to a DCT function 455, or outputs the inter pixel data to a subtractor 445, where a difference relative to an output from a motion compensation function 440 (responsive to the MV decoder 425) is provided to the DCT function 455.

The DCT coefficients are provided from the DCT function 455 to the quantisation function 340, and the quantised DCT data is then provided to the AC/DC (DCT coefficient) prediction function 350, where AC and DC residuals of the current MB are generated. These residuals of DCT coefficients are entropy coded. The output data is provided to the scan/run-length coding function 360, and the output thereof is provided to the variable length encoding function 370 to obtain the MPEG-4 compliant bitstream.

The quantised DCT coefficients are also output from the quantisation function 340 to an inverse quantisation function 495, the output of which is provided to an inverse DCT function 490, the output of which is summed at an adder 485 with the output of the motion compensation function 440. The output of the adder 485 is provided to a buffer 480, and subsequently to the motion compensation function 440.

The header decoding function 304 and look-up table 308 and path 312 operate as discussed in connection with FIG. 3 to control the re-encoding to the MPEG-4 format at functions 340-370.

4. Implementation of the Format Transcoder

This section explains the implementation of the format transcoding, e.g., as implemented in FIGs 3 and 4, discussed above, and FIG. 5, to be discussed later. Minor implementation details (e.g., systems-related details such as the use of time stamps and the like) that are not specifically discussed should be apparent to those skilled in the art.

In a particular implementation, the transcoders of the present invention can be used to convert a main-profile, main-level (MP@ML) MPEG-2 bitstream into a main-profile MPEG-4 bitstream. It is assumed that the MPEG-2 bitstream is coded in frame picture structure with B-picture coding (no dual prime prediction). Generally, the same coding mode which is used in MPEG-2 coding should be maintained. This mode is likely to be optimum in MPEG-4 and hence avoids the complexity of the mode decision process. The transparency pattern in MPEG-4 is always 1 (one rectangular object with the same size of VOP in one VOP). That is, MPEG-4 allows an arbitrarily shaped object which is defined by a nonzero transparency pattern. This feature does not exist in MPEG-2 so we can safely assume that all transparency patterns of the transcoding object is one.

4.1 MPEG-2 bitstream reader

A transcoder in accordance with the invention obtains the bitstream header, DCT coefficients and MVs from the MPEG-2 bitstream. This information is mixed together in the bitstream. Both MPEG-2 and MPEG-4

bitstreams adopt a hierarchical structure consisting of several layers. Each layer starts with the header following by a multiple of its sublayer. In this implementation, as shown in Table 1, the MPEG-2 layer has a direct translation into the MPEG-4 layer, except the slice layer in MPEG-2, which is not used in MPEG-4. DC coefficients and predicted MVs in MPEG-4 are reset at the blocks that start the slice.

However, some MPEG-4 headers are different from MPEG-2 headers, and vice versa. Fortunately, the restrictions in MPEG-2 and MPEG-2 header information are sufficient to specify a MPEG-4 header. Tables 2 through 6 list MPEG-4 headers and their relation to a MPEG-2 header or restriction at each layer.

Table 1. Relationship between MPEG-2 and MPEG-4 layers

MPEG-2	MPEG-4
Video Sequence	Video Object Sequence (VOS) / Video Object (VO)
Sequence Scalable Extension	Video Object Layer (VOL)
Group of Picture (GOP)	Group of Video Object Plane (GOV)
Picture	Video Object Plane (VOP)
Macroblock	Macroblock

Table 2. MPEG-4 header and its derivation (VOS and VO)

Header	Code	Comment
Visual_object_sequence_start_code	000001B0	Initiate a visual session
Profile_and_level_indication	00110100	Main Profile/Level 4

Visual_object_sequence_end_code	00001B1	Terminate a visual session
Visual_object_start_code	00001B5	Initiate a visual object
Is_visual_object_identifier	0	No version identification of priority needs to be specified
Visual_object_type	0001	Video ID
Video_object_start_code	0000010X-0000011X	Mark a new video object
Video_signal_type	Derived from MPEG-2	Corresponds to MPEG-2 sequence_display_extension_id
Video_format	Same as MPEG-2	Corresponds to MPEG-2 sequence_display_extension_id
Video_range	Derived from MPEG-2 '	Corresponds to MPEG-2 sequence_display_extension_id
Colour_description	Same as MPEG-2	Corresponds to MPEG-2 sequence_display_extension_id
Colour_primaries	Same as MPEG-2	Corresponds to MPEG-2 colour_description
Transfer_characteristics	Same as MPEG-2	Corresponds to MPEG-2 colour_description
Matrix_coefficients	Same as MPEG-2	Corresponds to MPEG-2 colour_description

Table 3. MPEG-4 header and its derivation (VOL)

Header	Code	Comment
Video_object_layer_start_code	0000012X	Mark a new video object layer
Random_accessible_vol	0	Allow non-intra coded VOP
Video_object_type_identification	00000100	Main object type
Is_object_type_identifier	0	No version identification of priority needs to be specified
Aspect_ratio_info	Same as MPEG-2	Corresponds to MPEG-2 aspect_ratio_information
Par_width	Same as	Corresponds to MPEG-2

	MPEG-2	vertical_size
Par_height	Same as MPEG-2	Corresponds to MPEG-2 horizontal_size
Vol_control_parameters	Same as MPEG-2	Correponds to MPEG-2 extension_start_code_identifier (sequence extension)
Chroma_format	Same as MPEG-2	Corresponds to MPEG-2 chroma_format
Low_delay	Same as MPEG-2	Corresponds to MPEG-2 low_delay
Vbv_parameters	Recomputed	Follow MPEG-4 VBV spec.
Video_object_layer_shape	00	Rectangular
Vop_time_increment_resolution	Recomputed	See Table 7
Fixed_vop_rate	1	Indicate that all VOPs are coded at a fixed rate
Fixed_vop_time_increment	Recomputed	See Table 7
Video_object_layer_width	Same as MPEG-2	Correpond to display_vertical_size
Video_object_layer_height	Same as MPEG-2	Correspond to display_horizontal_size
Interlaced	Same as MPEG-2	Correspond to progressive_sequence
Obmc_disable	1	Disable OBMC
Sprite_enable	0	Indicate absence of sprite
Not_8_bit	Derived from MPEG-2	Corresponds to MPEG-2 intra_dc_precision
Quant_type	1	MPEG quantization
Complexity_estimation_disable	1	Disable complexity estimation header
Resync_marker_disable	1	Indicate absence of resync_marker
Data_partitioned	0	Disable data partitioning
Reversible_vlc	0	Disable reversible vlc
Scalability	0	Indicate that the current layer is used as base-layer

Table 4. MPEG-4 header and its derivation (VOP)

Header	Code	Comment
Vop_start_code	000001B6	Mark a start of a video object plane
Vop_coding_type	Same as MPEG-2	Corresponds to MPEG-2 picture_coding_type
Modulo_time_base	Regenerated	Follow MPEG-4 spec.
Vop_time_increment	Regenerated	Follow MPEG-4 spec.
Vop_coded	1	Indicate that subsequent data exists for the VOP
Vop_rounding_type	0	Set value of rounding_control to '0'
Change_conversion_ratio_disable	1	Assume that conv_ratio is '1' for all macroblocks
Vop_constant_alpha	0	Not include vop_constant_alpha_value in the bitstream
Intra_dc_vlc_thr	0	Use intra DC vlc for entire VOP
Top_field_first	Same as MPEG-2	Corresponds to MPEG-2 top_field_first
Alternate_vertical_scan_flag	Same as MPEG-2	Corresponds to MPEG-2 to alternate_scan
Vop_quant	Derived from MPEG-2	Corresponds to MPEG-2 quantiser_scale_code
Vop_fcode_forward	Same as MPEG-2	See section 4.3
Vop_fcode_backward	Same as MPEG-2	See section 4.3

Table 5. MPEG-4 header and its derivation (macroblock and MV)

Header	Code	Comment
Not_coded	Derived from MPEG-2	Corresponds to MPEG-2 macroblock_address_increment

Mcbpc	Derived from MPEG-2	Corresponds to MPEG-2 macroblock_type
Ac_pred_flag	0	Disable intra AC prediction
Cbpy	Derived from MPEG-2	See section 4.2
Dquant	Derived from MPEG-2	See section 4.2
Modb	Derived from MPEG-2	Corresponds to macroblock_type
Mb_type	Derived from MPEG-2	Corresponds to macroblock_type
Cbpb	Derived from MPEG-2	See section 4.2
Dbquant	Derived from MPEG-2	See section 4.2
Horizontal_mv_data	Derived from MPEG-2	Corresponds to MPEG-2 motion_code[r][s][0]
Vertical_mv_data	Derived from MPEG-2	Corresponds to MPEG-2 motion_code[r][s][1]
Horizontal_mv_residual	Derived from MPEG-2	Corresponds to MPEG-2 motion_residual[r][s][0]
Vertical_mv_residual	Derived from MPEG-2	Corresponds to MPEG-2 motion_residual[r][s][1]

Table 6. MPEG-4 header and its derivation (block and interlaced information)

Header	Code	Comment
Dct_dc_size_luminance	Same as MPEG-2	Corresponds to MPEG-2 dct_dc_size_luminance
Dct_dc_differential	Same as MPEG-2	Correspond to dct_dc_differential
Dct_dc_size_chrominance	Same as MPEG-2	Corresponds to MPEG-2 dct_dc_size_chrominance
DCT_coefficient	Derived from MPEG-2	See section 4.2
DCT_type	Same as MPEG-2	Corresponds to MPEG-2 DCT_type
Field_prediction	Same as MPEG-2	Corresponds to MPEG-2 frame_motion_type
Forward_top_field_reference	Same as MPEG-2	Corresponds to MPEG-2 motion_vertical_field_select[0][0]

Forward_bottom_field_reference	Same as MPEG-2	Corresponds to MPEG-2 motion_vertical_field_select[1][0]
Backward_top_field_reference	Same as MPEG-2	Corresponds to MPEG-2 motion_vertical_field_select[0][1]
Backward_bottom_field_reference	Same as MPEG-2	Corresponds to MPEG-2 motion_vertical_field_select[1][1]

Table 7. Mapping of frame_rate_code in MPEG-2 to vop_time_increment_resolution and fixed_vop_time_increment in MPEG-4.

Frame_rate_code	Vop_time_increment_resolution	Fixed_vop_time_increment
0001	24,000	1001
0010	24	1
0011	25	1
0100	30,000	1001
0101	30	1
0110	50	1
0111	60,000	1001
1000	60	1

MV data is stored in the macroblock layer. Up to four MVs are possible for each macroblock. Moreover, a MV can be of either field or frame type and have either full pixel or half pixel resolution. The MPEG-2 MV decoding process is employed to determine motion_code (VLC) and motion_residual (FLC) and, hence, delta. Combined with predictive MV, delta gives the field/frame MV. The MV for skipped macroblocks is set to zero.

DCT data is stored in the block layer. It is first decoded from the bitstream (VLC), inverse scanned

using either zigzag or alternate scanning pattern, and then inverse quantized. The intra DC coefficient is determined from dct_dc_differential and the predictor (the predictor is reset according to the MPEG-2 spec). DCT coefficients in a skipped macroblock are set to zero.

4.2 Texture coding

A transcoder in accordance with the invention reuses DCT coefficients (for inter frame). The following guidelines should be used:

1. q_scale_type = 1 (linear scale) is used in MPEG-2 quantization.

2. The MPEG quantization method should only be used (not H.263) in MPEG-4 quantization to reduce a mismatch between MPEG-2 and MPEG-4 reconstructed frame (drift).

3. A differential value of MPEG-2 QP determines dquant in MPEG-4. Dquant is set to ±2 whenever the differential value is greater than ±2. dquant is a 2-bit code which specifies a change in the quantizer, quant, for I- and P-VOPs.

4. The quantization matrix should be changed following the change of matrix in the MPEG-2 bitstream.

5. The transcoder has the flexibility of enabling an alternate vertical scanning method (for interlaced sequence) at the VOL level.

6. Intra AC/DC prediction (which involves scaling when the QP of the current block is not the same as that of the predicted block) should be turned

off at a macroblock level to reduce complexity and mismatch in AC quantization.

7. Higher efficiency can be obtained with the use of intra_dc_vlc_thr to select the proper VLC table (AC/DC) for coding of intra DC coefficients, e.g., as a function of the quantization parameter (except when intra_dc_vlc_thr is either 0 or 7 - these thresholds will force the use of the intra DC or AC table regardless of the QP).

8. A skipped macroblock is coded as not_coded macroblock (all DCT coefficients are zero).

9. Cbpy and cbpc (CBP) are set according to code_block_pattern_420 (CBP_420). Note that there is a slight discrepancy between CBP in MPEG-4 and CBP_420 in MPEG-2 for an intra macroblock. Specifically, when CBP_420 is set, it indicates that at least one of the DCT coefficients in that block is not zero. CBP contains similar information except it does not corresponds to a DC coefficient in an intra macroblock (also depending on intra_dc_vlc_thr). Hence, it is possible that CBP is not zero when CBP_420 is zero in an intra macroblock (this case can happen in an I-VOP and P-VOP, but not B-VOP).

There are three sources of loss in texture coding, namely QP coding, DC prediction and nonlinear scaler for DC quantization. MPEG-4 uses differential coding to code a QP. MPEG-2 allows all possible 32 QP values at the expense of 5 bits. However, the differential value can take up to ±2 (in QP value units) and, hence,

a differential value greater than ±2 is loss. This loss can be minimized by limiting the QP fluctuation among the macroblock in the MPEG-2 rate control algorithm. All intra macroblocks perform adaptive DC prediction, which may take a different prediction from the previous macroblock (MPEG-2 DC prediction) thereby causing a different DC residual for the quantization. DC coefficients of all intra macroblocks in MPEG-4 are also quantised in a different manner from MPEG-2 because of the nonlinear scaler. Therefore, quantised DC coefficients for MPEG-2 and MPEG-4 coding are likely to be different for an intra macroblock.

4.3 MV coding

The transcoder encodes MVs into an MPEG-4 format. However, there is no error involved in transcoding a MV from MPEG-2 to MPEG-4 since MV coding is a lossless process. The following constraints are imposed on a MPEG-4 encoder:

1. Unrestricted motion compensation mode is disabled, which means no MV pointing outside the boundary of the frame.

2. Advanced prediction mode is employed. A different predictor (a median value) is used in an MPEG-4 bitstream, but a MV for 8x8 pels block is not. That is, advanced prediction mode allows 8x8 MV and nonlinear (median filter) predictor. Only a nonlinear predictor is used in our format transcoder (we still keep a 16x16 MV).

3. Direct mode is not allowed in an MPEG-4 bitstream, which means there are only four MV types for a B-VOP, i.e., 16×16 forward and backward vectors and 16×8 forward and backward field vectors.

4. Field motion compensation is applied whenever a 16×8 field vector is used (maintain mode).

5. A skipped macroblock is coded as not_coded macroblock (motion compensation with zero MV).

6. Single f_code is allowed in MPEG-4. Therefore, the larger f_code in MPEG-2 between the two directions (vertical, horizontal) is converted to f_code in MPEG-4 based on the following relationship: f_code(MPEG-4) = f_code(MPEG-2)-1.

7. A padding process is not used since the texture for the entire reference frame is known.

8. Field motion compensation is used whenever dual prime arithmetic is activated. Vector parities (field of the reference and field of the predicting frame) are preserved. Field MVs are generated according to vector[0][0][1:0] which is coded in the MPEG-2 bitstream. When prediction of the same parity is used (e.g., top field to top field, or bottom field to bottom field), both field MVs are vector[0][0][1:0]. When prediction of the odd parity is used (e.g., top field to bottom field, or bottom field to top field), the top field MV uses vector[2][0][1:0] and the bottom field MV uses vector[3][0][1:0]. Vector[r][0][0:1] for r=2,3 can computed as follows:

(a) Vector[r][0][0] = (vector[0][0][0] × m[parity_ref][parity_pred]//2) + dmvector[0].

(b) Vector[r][0][1] = (vector[0][0][1] × m[parity_ref][parity_pred]//2) + e[parity_ref][parity_pred] + dmvector[1].

Note that (m[parity_ref][parity_pred] and e[parity_ref][parity_pred] are defined in Table 7-11 and 7-12, respectively in the MPEG-2 specification (ISO/IEC 13818-2).

Moreover, "r" denotes the order of the MV, e.g., first, second, etc. r=0 denotes to the first set of MV, and r=1 denotes the second set of MV. Dual prime prediction uses r=2 and r=3 to identify two extra sets of MVs.

"//" denotes integer division with rounding to the nearest integer.

4.4 Coding of intra MB in B-VOP

Additional conversion is necessary when coding an intra MB in a B-frame of a MPEG-2 bitstream (e.g., as shown in FIG. 4). MPEG-4 replaces intra mode with direct mode for B-VOP and hence an intra MB in B-frame has to be coded differently in the MPEG-4 syntax. There are two practical solutions to this problem.

The first solution employs the architecture similar to the front-to-back transcoder of FIG. 3 (no buffer for the entire reference frame). MC is performed against the previous MB (or previous MB without compensating texture residual with the expense of the extra memory with the size of one MB) in the

same VOP under the assumption that this MB is close enough to its reference MB (its uncompensated version). The MV for the intra MB equals the MV of the previous MB offset by its MB distance.

The second solution uses the architecture similar to the one shown in FIG. 4. It keeps the reference frame for all I and P-VOPs. Note that MC has to be performed on all P-VOPs in this solution. The MV for the intra MB is the same as the predicted MV (median of its three neighbors) and MC is performed against the reference MB pointed by the derived MV.

5. Video downscaling in the compressed domain

Generally, video downscaling and size transcoding have the same meaning. Downsampling means sub-sampling with an anti-aliasing (low pass) filter, but subsampling and downsampling are used interchangeably herein.

Size transcoding becomes computationally intensive when its input and output are in the compressed domain. A video downscaling process which limits its operations in the compressed domain (and, in effect, avoids decoding and encoding processes) provides a much reduced complexity. However, two new problem arises with downscaling in the compressed domain, i.e., downsampling of DCT coefficients and MV data.

Recently, video downscaling algorithms in the compressed domain have been discussed, but they do not address the complete transcoding between MPEG-2 and

MPEG-4, which includes field-to-frame deinterlacing. The present invention addresses this problem.

Subsection 5.1 and 5.2 provide solutions to two new problems in the downsampling process. The implementation of a proposed size transcoder in accordance with the invention is described in section 6 and FIGs 5 and 6.

5.1 Subsampling of DCT block

In frame-based video downscaling, it is necessary to merge four 8x8 DCT blocks into a new 8x8 DCT block (specific details involving a field block will be described later). Moreover, the output block should be a low pass version of the input blocks. This process is carried out in the spatial domain by multiplying the input matrix with a subsampling matrix (preferably with a low pass filter). Multiplication by a subsampling matrix in the spatial domain is equivalent to multiplication by DCT coefficients of a subsampling matrix in the DCT domain because of the distributive property of the orthogonal transform. However, the number of operations (computations) in the downsampling process in the DCT domain for some downsampling filters can be as high as the total number of operations of its counterpart in the spatial domain. The solution to this problem is to employ a downsampling matrix which is sparse (e.g., a matrix that has relatively few non-zero values, e.g., approximately 30% or less).

A sparse downsampling matrix may be based on the orthogonal property between the DCT basis vector and

the symmetry structure of the DCT basis vector. One approach, discussed in R. Dugad and N. Ahuja, "A Fast Scheme For Downsampling And Upsampling In The DCT Domain," International Conference on Image Processing (ICIP) 99, incorporated herein by reference, takes the lower 4x4 DCT coefficients from four processing blocks, applies a 4x4 IDCT to each DCT subblock, forms a new 8x8 pixel block and applies an 8x8 DCT to obtain an output block. The downsampling matrix can be pre-calculated since the downsampling process is fixed. By splitting the 8x8 DCT matrix into left and right halves, about half of the downsampling matrix values are zero because of the orthogonality between the column of the 4x4 IDCT matrix and the row of both left and right 8x4 DCT matrices. This operation (one dimension) can be written mathematically as:

$$B = Tb = T\begin{bmatrix} b_1 \\ \cdots \\ b_2 \end{bmatrix} = \begin{bmatrix} T_L & \vdots & T_R \end{bmatrix}\begin{bmatrix} T_4^t B_1 \\ \cdots \\ T_4^t B_2 \end{bmatrix} = T_L T_4^t B_1 + T_R T_4^t B_2$$

where b is a 8x1 spatial input vector, B is its corresponding 8x1 DCT vector, b_1 and b_2 are subsampled 4x1 vectors, B_1 and B_2 are lower 4x1 DCT vectors, T is the 8x8 DCT transform matrix, T_4 is the 4x4 DCT transform matrix, T_L and T_R are left and right half of T. The superscript "t" denotes a matrix transpose. Dugad's algorithm also employs the symmetry property of the DCT basis vector to reduce the complexity of the downsampling process. $T_L T_4^t$ and $T_R T_4^t$ are identical in

terms of magnitude ($T_L T_4^t(i,j)=(-1)^{i+j} T_R T_4^t(i,j), 0\le i\le 7, 0\le j\le 3$) since odd rows of T are anti-symmetric and even rows of T are symmetric. "i" is a matrix row index, and "j" is a matrix column index. Hence, both $T_L T_4^t$ and $T_R T_4^t$ can be calculated based on the same components, i.e., a symmetrical part, E, (index which $i+j$ is even) and an anti-symmetrical part, O, (index which $i+j$ is odd) ($T_L T_4^t = E+O$ and $T_R T_4^t = E-O$). This arrangement effectively reduces the number of multiplications by a factor of two when the downsampling process is done as:

$$B = T_L T_4^t B_1 + T_R T_4^t B_2 = (E+O)B_1 + (E-O)B_2 = E(B_1+B_2) + O(B_1-B_2)$$

Implementation of Dugad's method to convert four field blocks into one frame block is not as simple. An extension of the downsampling process in this scenario (one dimension) can be written as:

$$B = T(S_T T_4^t B_T + S_B T_4^t B_B)$$

where B_T and B_B are the lower 4x1 field vectors, S_T and S_B are DCT values of an 8x4 deinterlacing matrix corresponding to its top, s_T, and bottom, s_B, field block, respectively. Elements of s_T, $s_T(i,j)=1$ if ($j=2i$, $0\le i\le 3$) and $s_T(i,j)=0$ otherwise. Elements of s_B, $s_B(i,j)=1$ if ($j=2i+1$, $0\le i\le 3$) and $s_B(i,j)=0$ otherwise.

This is a modification of Dugad's algorithm for downsampling and deinterlacing in accordance with the present invention.

The operations of downscaling and the deinterlacing process are more complex since S and T are not orthogonal to each other and, hence, the downsampling matrix is not sparse. C. Yim and M.A. Isnardi, "An Efficient Method For DCT-Domain Image Resizing With Mixed Field/Frame-Mode Macroblocks," IEEE Trans. Circ. and Syst. For Video Technol., vol. 9, pp.696-700, Aug. 1999, incorporated herein by reference, propose an efficient method for downsampling a field block. A low pass filter is integrated into the deinterlacing matrix in such a way that the downsampling matrix (S=0.5[I_8 I_8]) is sparse.

I_8 denotes an 8x8 identity matrix, and [I_8 I_8] denotes a 16x8 matrix that comprises a concatenation of the two identity matrixes. The identity matrix, of course, has all ones on the diagonal and all zeroes elsewhere.

The method starts with four 8x8 IDCT field blocks, then applies the downsampling matrix, S, and performs an 8x8 DCT to obtain the output block. Note that an 8x8 IDCT is used in this method instead of a 4x4 IDCT. This operation can be shown mathematically (in one dimension) as:

$$D = TS\begin{bmatrix} T^t & 0 \\ 0 & T^t \end{bmatrix} = \frac{1}{2}T\begin{bmatrix} I_8 & \vdots & I_8 \end{bmatrix}\begin{bmatrix} T^t & 0 \\ 0 & T^t \end{bmatrix} = \frac{1}{2}\begin{bmatrix} I_8 & \vdots & I_8 \end{bmatrix}$$

.

5.2 Subsampling of MV data

ME is the bottleneck of the entire video encoding process. It is hence desirable to estimate a MV of the resized MB by using MVs of four original MBs without actually performing ME (assuming that all MBs are coded in inter mode). Note that, if an MPEG-2 bitstream is assumed, subsampling of MV data takes MVs of four MBs since each MB has one input (only an MPEG-4 bitstream can have a MV for every block). The simplest solution is to average four MVs together to obtain the new MV but it gives a poor estimate when those four MVs are different. B. Shen, I.K. Sethi and B. Vasudev, "Adaptive Motion-Vector Resampling For Compressed Video Downscaling," IEEE Trans. Circ. and Syst. For Video Technol., vol. 9, pp. 929-936, Sep. 1999, show that a better result can be obtained by giving more weight to the worst predicted MV. A matching accuracy, A, of each MV is indicated by the number of nonzero AC coefficients in that MB. By using the Shen et al. technique, the new MV for the downscaled ME can be computed as:

$$MV' = \frac{\frac{1}{2}\sum_{i=1}^{4} MV_i A_i}{\sum_{i=1}^{4} A_i}$$

M.R. Hashemi, L. Winger and S. Panchanathan, "Compressed Domain Motion Vector Resampling For Downscaling Of MPEG Video," ICIP 99, propose a nonlinear method to estimate the MV of the resized ME. Similar to the algorithm in Shen et al., Hashemi's

technique uses spatial activity of the processing MBs to estimate the new MV. A heuristic measurement, called Maximum Average Correlation (MAC) is employed in Hashemi's method to identify one of the four original MVs to be the output MV. By using the MAC, the new MV for the downscaled MB can be computed as::

$$MV = \max \sum_{i=1}^{4} A_i \rho^{d_i}$$

where ρ is the spatial correlation and is set to 0.85, and d_i is the Euclidean distance between the ith input MV (MV_i) and the output MV.

6. Implementation of the size transcoder

FIG. 5 illustrates a size transcoder in accordance with the invention. B frames may be present in the input bitstream, but are discarded by the transcoder and therefore do not appear in the output bitstream.

In the transcoder 500, a MV scaling function 510, DCT scaling function 520, and spatial scaling function 540 are added. Switches 530 and 535 are coordinated so that, in a first setting, an output of the DCT function 455 is routed into the quantisation function 340, and the switch 535 is closed to enable an output of the spatial scaling function 540 to be input to the adder 445. In a second setting of the switches 530 and 535, an output of the DCT scaling function 520 is routed into the quantisation function 340, and the switch 535 is open.

The transcoder 500 converts an MPEG-2 bitstream into an MPEG-4 bitstream which corresponds to a smaller size video, e.g., from ITU-R 601 (720x480) to SIF (352x240).

To achieve a bandwidth requirement for the MPEG-4 bitstream, the transcoder 500 subsamples the video by two in both the horizontal and vertical directions (at the spatial scaling function 540) and skips all B-frames (at temporal scaling functions 545 and 546), thereby reducing the temporal resolution accordingly. Note that the temporal scaling function 546 could alternatively be provided after the DCT scaling function 520. Skipping of B-frames before performing downscaling reduces complexity.

Moreover, a low pass filter (which can be provided in the spatial scaling function 540) prior to subsampling should result in improves image quality.

The invention can be extended to include other downsampling factors, and B-VOPs, with minor modifications. Specifically, changes in MV downscaling and mode decision are made. MV downscaling for B-VOP is a direct extension of what was discussed to include the backward MV. The mode decision for B-VOP can be handled in a similar way as in the P-VOP (e.g., by converting uni-directional MV into bi-directional MV as in converting intra MB into inter MB in a P-VOP).

Below, we discuss six problems that are addressed by the size transcoder 500. We also assume that the input video is 704x480 pixel resolution, and coded with

an MP@ML MPEG-2 encoder, and the desired output is simple profile MPEG-4 bitstream which contains SIF progressive video (with a frame rate reduction by N). However, the invention can be extended to other input and output formats and resolutions as well.

6.1 Progressive Video MV downscaling (luma)

This problem appears when all four MBs are coded as inter, and use frame prediction. Each MV in those MBs is downscaled by two in each direction (horizontal and vertical) to determine the MV of four blocks in MPEG-4 (MPEG-4 allows one MV per 8x8 block). The scaled MVs are then predictively encoded (using a median filter) using the normal MPEG-4 procedure.

Note that each MB (comprising four blocks) has to be coded in the same mode in both MPEG-2 and MPEG-4. With video downscaling, the output MB (four blocks) corresponds to four input MBs.

6.2 Interlaced Video MV downsampling (luma)

This problem exists when all four MBs are coded as inter and use field prediction. We need to combine two field MVs in each MB to get a frame MV of the resized block. Instead of setting the new MV based on the spatial activity, the proposed transcoder picks the new MV based on its neighbors' MVs. The MVs of all eight surrounding MBs are used to find a predictor (field MVs are averaged in case of MB with field prediction). The median value from these eight MVs becomes a predictor, and the field MV of the current MB, which is closer in

terms of Euclidean distance, is scaled by two in the horizontal direction to become the new MV.

6.3 MV downsampling (chroma)

This problem happens when all four MBs are coded as inter, and use either frame or field prediction (MPEG-4 treats both prediction mode in the same way for a chroma block). The process follows the MPEG-4 method to obtain a chroma MV from a luma MV, i.e., a chroma MV is the downscaled version of the average of its four corresponding, 8x8 luma MVs.

6.4 DCT downsampling (luma progressive, chroma)

This problem occurs when all four luma MBs are coded as intra or inter, and use frame MB structure, and their eight chroma blocks (four for Cr and four for Cb) use either frame or field structure). Dugad's method is used to downscale the luma and chroma DCT blocks by a factor of two in each direction.

6.5 Interlaced DCT downsampling (luma)

This problem arrives in one of two ways. First, its associated MB uses field prediction and second, its associated MB uses frame prediction. In either case, we want to downscale four 8x8 field DCT blocks (two for the top field, and two for the bottom field) into one 8x8 frame DCT block. The solution for the first case is to use the same field DCT block as the one chosen for MC. The second case involves deinterlacing and we propose a combination of the Dugad and Yim methods, discussed above.

Specifically, the transcoder first downscales four field blocks in the vertical direction (and at the same time performs deinterlacing) based on the Yim algorithm to obtain two frame blocks. The transcoder then downscales these two frame blocks in the horizontal direction to get the output block using the Dugad algorithm.

This is illustrated in FIG. 6, where four 8x8 coefficient field-mode DCT blocks are shown at 600, two 8x8 frame-mode DCT blocks are shown at 610, and one 8x8 frame-mode DCT block is shown at 620.

The procedure for DCT downscaling in accordance with the invention can be summarized as follows:

1. Form the 16x16 coefficient input matrix by combining four field blocks together as shown at 600.

2. For vertical downscaling and filtering, apply a low pass (LP) filter D according to Yim's algorithm to every row of the input matrix. The LP input matrix is now 16x8 pixels, as shown at 610.

3. Form B_1 and B_2 8x8 matrices from the LP matrix ($\left[B_1 \vdots B_2\right]$).

4. Perform a horizontal downscaling operation according to Dugad's algorithm to every column of B_1 and B_2 to obtain the output matrix (8x8) (620) as follows:

$$B = B_1(T_L T_4^t)^t + B_2(T_R T_4^t)^t = (B_1 + B_2)E + (B_1 - B_2)O$$

where E and O denote even and odd rows as discussed above.

In particular, a horizontal downsampling matrix composed of odd "O" and even "E" matrices as follows may be used (ignoring the scaling factor):

E = [e(0) 0 0 0,
0 e(1) 0 e(2),
0 0 0 0,
0 e(3) 0 e(4),
0 0 e(5) 0,
0 e(6) 0 e(7),
0 0 0 0,
0 e(8) 0 e(9)].

O = [0 0 0 0,
o(0) 0 o(1) 0,
0 o(2) 0 0,
o(3) 0 o(4) 0,
0 0 0 0,
o(5) 0 o(6) 0,
0 0 0 o(7),
o(8) 0 o(9) 0].

The coefficients as follows can be used:

e(0) = 4 o(0) = 2.56915448
e(1) = 0.831469612 o(1) =-0.149315668
e(2) = 0.045774654 o(2) = 2
e(3) = 1.582130167 o(3) =-0.899976223
e(4) =-0.195090322 o(4) = 1.026559934
e(5) = 2 o(5) = 0.601344887
e(6) =-0.704885901 o(6) = 1.536355513
e(7) = 0.980785280 o(7) = 2

e(8) = 0.906127446 o(8) =-0.509795579

e(9) = 1.731445835 o(9) =-0.750660555.

Essentially, the product of a DCT matrix which is sparse is used as the downsampling matrix.

The technique may be extended generally for 2:1 downsizing of an NxN block that comprises four N/2xN/2 coefficient field-mode blocks. Other downsizing ratios may also be accommodated.

6.6 Special cases

Special cases occur when all four MBs are not coded in the same mode (not falling in any of the five previous cases). We always assume that any intra or skipped MB among the other inter MBs are inter mode with zero MV. Field MVs are merged based on section 6.2 to obtain frame MV, and then we apply the techniques of section 6.1. MC is recommended to determine the texture of the intra block, which is treated as an inter block with a zero MV by the transcoder.

7. Conclusion

It should now be appreciated that the present invention provides a transcoder architecture that provides the lowest possible complexity with a small error. This error is generated in the MPEG-4 texture encoding process (QP coding, DC prediction, nonlinear DC scaler). These processes should be removed in the future profile of MPEG-4 to create a near-lossless transcoding system.

The invention also provides complete details of a size transcoder to convert a bitstream of ITU-R 601 interlaced video coding with MPEG-2 MP@ ML into a simple profile MPEG-4 bitstream which contains SIF progressive video suitable for a streaming video application.

For spatial downscaling of field-mode DCT blocks, it is proposed to combine vertical and horizontal downscaling techniques in a novel manner such that sparse downsampling matrixes are used in both the vertical and horizontal direction, thereby reducing computations of the transcoder.

Moreover, for MV downscaling, we propose using a median value from its eight neighboring MV. This proposal works better than algorithms in section 5.2 since our predicted MV go with the global MV. It also works well with an interlaced MB, which has only two MVs instead of 4 MVs per MB.

Although the invention has been described in connection with various preferred embodiments, it should be appreciated that various modifications and adaptations may be made thereto without departing from the scope of the invention as set forth in the claims.

What is claimed is:

1. A method for transcoding a pre-compressed input bitstream that is provided in a first video coding format, comprising the steps of:

recovering header information of the input bitstream;

providing corresponding header information in a second, different video coding format;

partially decompressing the input bitstream to provide partially decompressed data; and

re-compressing the partially decompressed data in accordance with the header information in the second format to provide an output bitstream.

2. The method of claim 1, wherein:

the first and second video coding formats comprise an MPEG-2 format and an MPEG-4 format, respectively.

3. The method of claim 1, wherein:

the first video coding format comprises MPEG-2 Main Profile at Main Level; and

the second video coding format comprises a simple profile MPEG-4 bitstream with Standard Intermediate Format (SIF) progressive video.

4. The method of claim 1, wherein:

the partially decompressed data comprises motion vectors and Discrete Cosine Transform (DCT) coefficients; and

the second format comprises at least one of a new mode decision, AC/DC prediction, and motion compensation.

5. The method of claim 1, wherein:

at least one look-up table is used to provide the corresponding header information in the second video coding format.

6. The method of claim 1, wherein:

downscaling is performed on the partially decompressed data by downsampling DCT coefficients and motion vector data thereof.

7. The method of claim 1, wherein:

2:1 downscaling is performed on at least one group of four field-mode Discrete Cosine Transform (DCT) blocks of the partially decompressed data by performing vertical downsampling and de-interlacing thereto to obtain a corresponding group of two frame-mode DCT blocks, and performing horizontal downsampling to the two frame-mode DCT blocks to obtain one frame-mode DCT block.

8. The method of claim 7, wherein:

the vertical downsampling also achieves low pass filtering of the four field-mode DCT blocks.

9. The method of claim 7, wherein:
the vertical and horizontal downsampling use respective sparse matrixes.

10. The method of claim 1, wherein:
in the recompressing step, a code (DQUANT) which specifies a change in a quantizer is set according to a differential value of a quantization parameter of the partially decompressed data.

11. The method of claim 1, wherein:
for re-compressing intra coded macroblocks, a coded block pattern (CBP) is set according to a corresponding value of the partially decompressed data.

12. The method of claim 1, wherein:
for re-compressing non-intra coded macroblocks, skipped macroblocks in the partially decompressed data are coded as not_coded macroblocks, where all Discrete Cosine Transform (DCT) coefficients have a zero value.

13. The method of claim 1, wherein:
in the recompressing step, predicted motion vectors in the partially decompressed data are reset according to the second format.

14. The method of claim 1, wherein:

in the recompressing step, dual prime mode macroblocks of the partially decompressed data are converted into field-coded macroblocks.

15. A method for performing 2:1 downscaling on video data, comprising the steps of:

forming at least one input matrix of NxN Discrete Cosine Transform (DCT) coefficients from the video data by combining four N/2xN/2 field-mode DCT blocks;

performing vertical downsampling and de-interlacing to the input matrix to obtain two N/2xN/2 frame-mode DCT blocks;

forming an NxN/2 input matrix from the two frame-mode DCT blocks; and

performing horizontal downsampling to the NxN/2 matrix to obtain one N/2xN/2 frame-mode DCT block.

16. The method of claim 15, wherein N=16.

17. The method of claim 15, wherein:

the vertical downsampling also achieves low pass filtering of the NxN input matrix.

18. The method of claim 15, wherein:

the vertical downsampling uses a sparse downsampling matrix.

19. The method of claim 18, wherein:

the sparse downsampling matrix=0.5[I_8 I_8], where I_8 is an 8x8 identity matrix.

20. The method of claim 15, wherein:

the horizontal downsampling uses a sparse downsampling matrix composed of odd "O" and even "E" matrices.

21. The method of claim 20, wherein:

the even matrix has the following form:

```
E = [e(0) 0    0    0,
     0    e(1) 0    e(2),
     0    0    0    0,
     0    e(3) 0    e(4),
     0    0    e(5) 0,
     0    e(6) 0    e(7),
     0    0    0    0,
     0    e(8) 0    e(9)]
```

where e(1) through e(9) are non-zero coefficients; and

the odd matrix has the following form:

```
O = [0    0    0    0,
     o(0) 0    o(1) 0,
     0    o(2) 0    0,
     o(3) 0    o(4) 0,
     0    0    0    0,
     o(5) 0    o(6) 0,
     0    0    0    o(7),
     o(8) 0    o(9) 0]
```

where o(1) through o(9) are non-zero coefficients.

22. An apparatus for transcoding a pre-compressed input bitstream that is provided in a first video coding format, comprising:

means for recovering header information of the input bitstream;

means for providing corresponding header information in a second, different video coding format;

means for partially decompressing the input bitstream to provide partially decompressed data; and

means for re-compressing the partially decompressed data in accordance with the header information in the second format to provide an output bitstream.

23. An apparatus for performing 2:1 downscaling on video data, comprising:

means for forming at least one input matrix of NxN Discrete Cosine Transform (DCT) coefficients from the video data by combining four N/2xN/2 field-mode DCT blocks;

means for performing vertical downsampling and de-interlacing to the input matrix to obtain two N/2xN/2 frame-mode DCT blocks;

means for forming an NxN/2 input matrix from the two frame-mode DCT blocks; and

means for performing horizontal downsampling to the NxN/2 matrix to obtain one N/2xN/2 frame-mode DCT block.

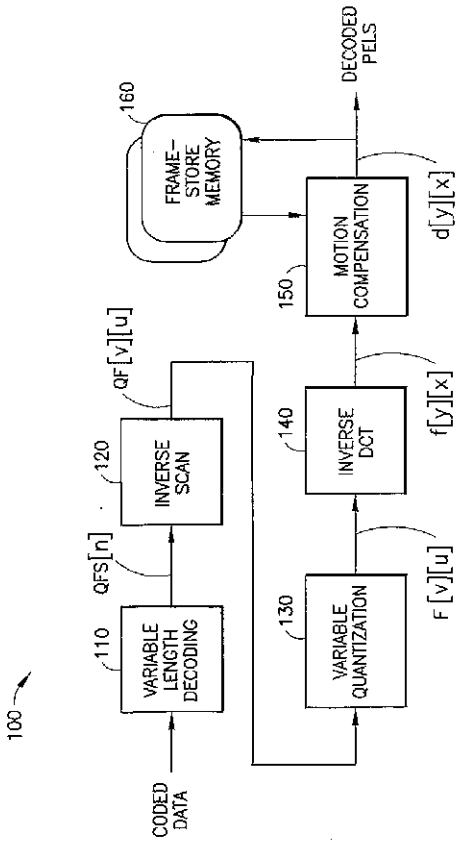


FIG.1
PRIOR ART

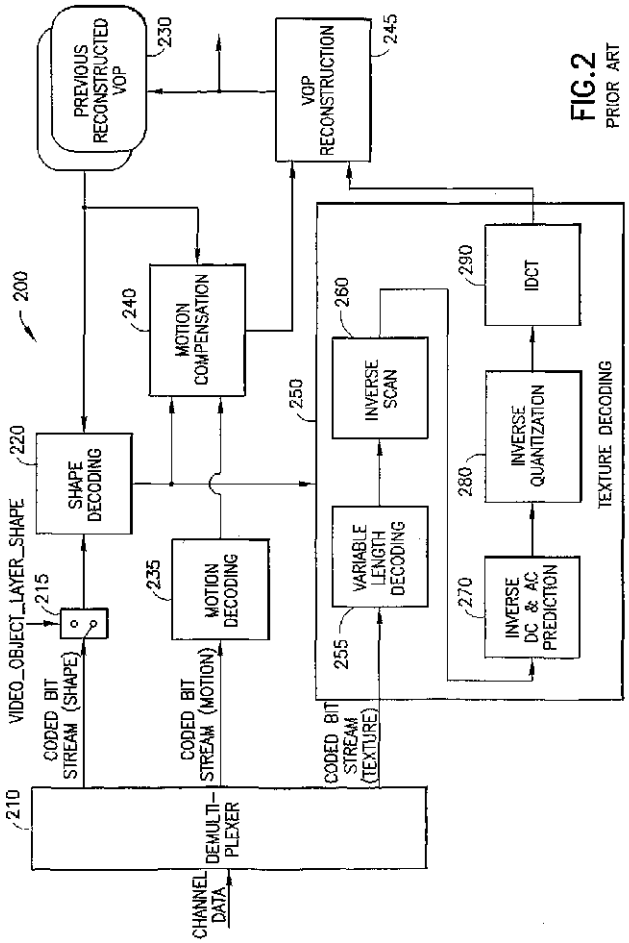


FIG.2
PRIOR ART

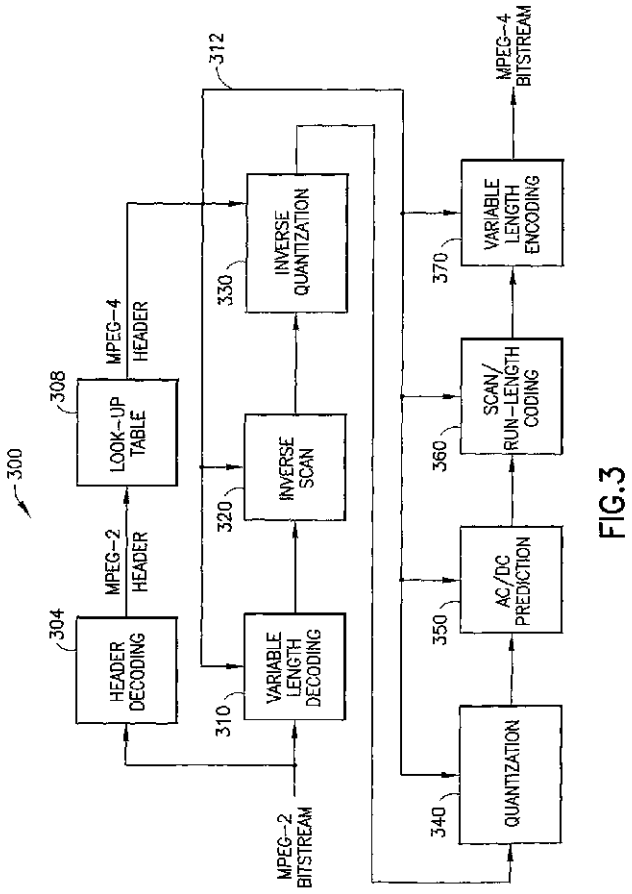

300
304
HEADER DECODING
MPEG-2 HEADER
308
LOOK-UP TABLE
MPEG-4 HEADER
310
VARIABLE LENGTH DECODING
320
INVERSE SCAN
330
INVERSE QUANTIZATION
312
MPEG-2 BITSTREAM
340
QUANTIZATION
350
AC/DC PREDICTION
360
SCAN/ RUN-LENGTH CODING
370
VARIABLE LENGTH ENCODING
MPEG-4 BITSTREAM

FIG.3

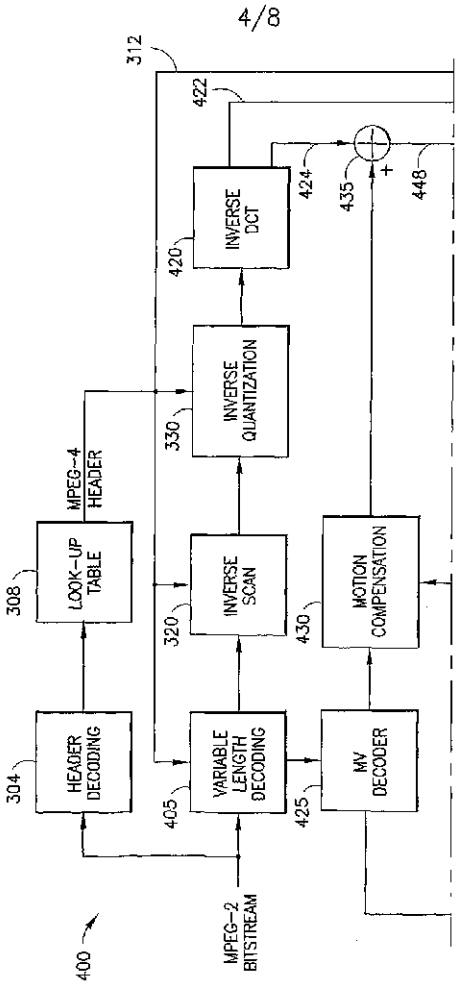


FIG.4A

FIG.4B

FIG.4

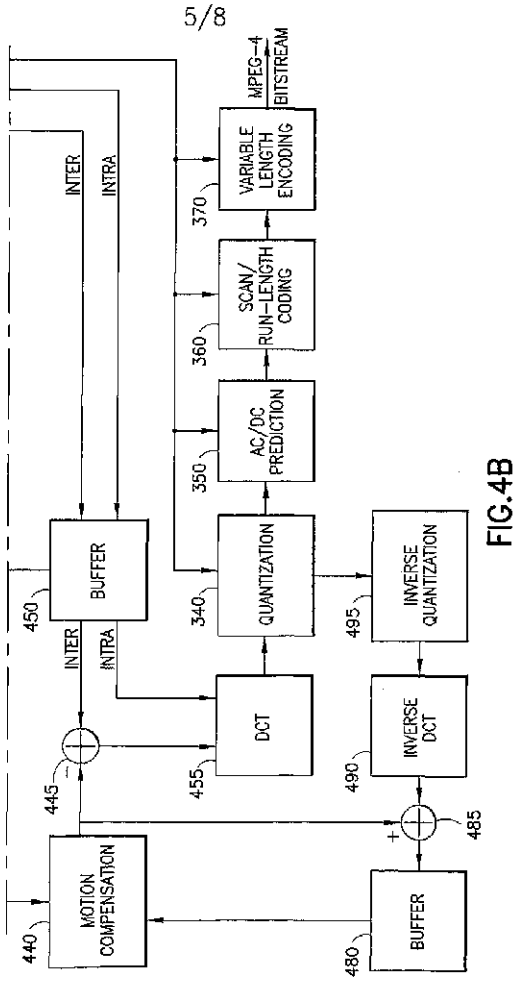

MPEG-4 BITSTREAM
VARIABLE LENGTH ENCODING
370
SCAN/ RUN-LENGTH CODING
360
AC/DC PREDICTION
350
QUANTIZATION
340
INVERSE QUANTIZATION
495
BUFFER
450
INTER
INTRA
DCT
455
INVERSE DCT
490
445
485
MOTION COMPENSATION
440
BUFFER
480

FIG.4B

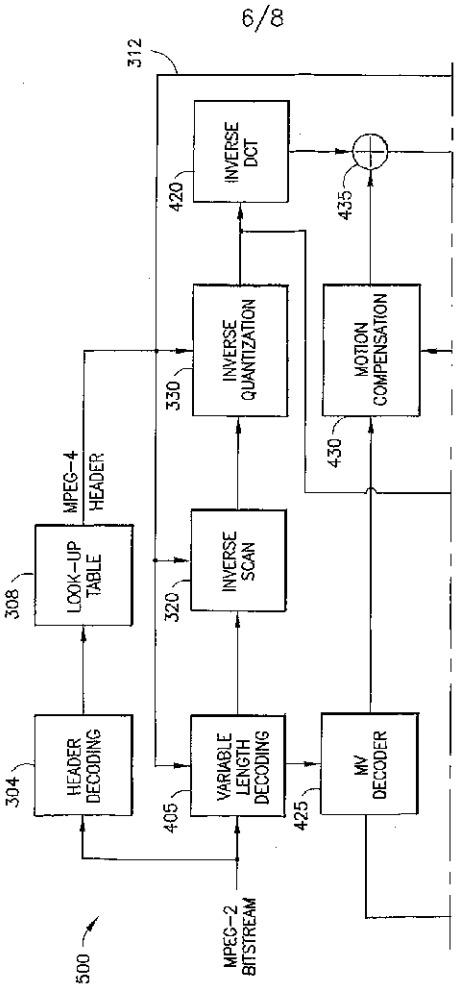


FIG.5A

FIG.5B

FIG.5

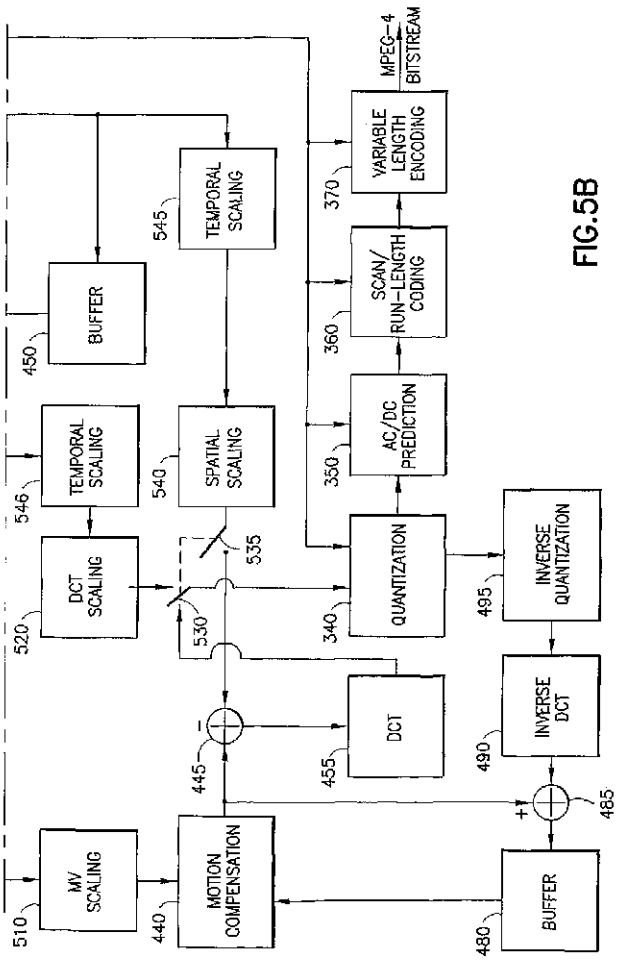

MPEG-4 BITSTREAM
VARIABLE LENGTH ENCODING
370
SCAN/ RUN-LENGTH CODING
360
AC/DC PREDICTION
350
QUANTIZATION
340
INVERSE QUANTIZATION
495
INVERSE DCT
490
DCT
455
TEMPORAL SCALING
545
SPATIAL SCALING
540
TEMPORAL SCALING
546
DCT SCALING
520
BUFFER
450
MV SCALING
510
MOTION COMPENSATION
440
BUFFER
480
535
530
445
485

FIG.5B

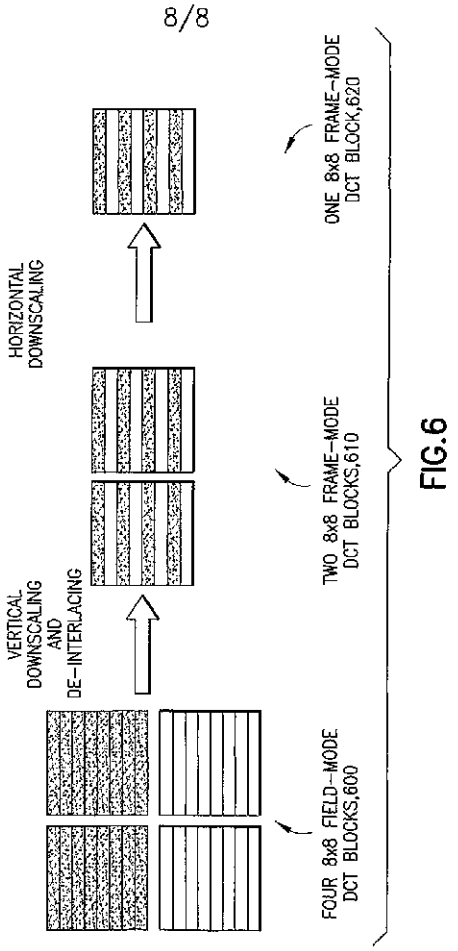

VERTICAL DOWNSCALING AND DE-INTERLACING
HORIZONTAL DOWNSCALING
FOUR 8x8 FIELD-MODE DCT BLOCKS,600
TWO 8x8 FRAME-MODE DCT BLOCKS,610
ONE 8x8 FRAME-MODE DCT BLOCK,620

FIG.6

【国際公開パンフレット（コレクトバージョン）】

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
13 December 2001 (13.12.2001)

PCT

(10) International Publication Number
WO 01/095633 A3

(51) International Patent Classification⁷: H04N 7/26

(21) International Application Number: PCT/US01/40811

(22) International Filing Date: 25 May 2001 (25.05.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
09/590,928 9 June 2000 (09.06.2000) US

(71) Applicant (*for all designated States except US*): GENERAL INSTRUMENT CORPORATION [US/US]; 101 Tournament Drive, Horsham, PA 19044 (US).

(72) Inventors; and

(75) Inventors/Applicants (*for US only*): PANUSOPONE, Krit [TH/US]; 9656 Carroll Canyon Road #14, San Diego, CA 92126 (US). CHEN, Xuemin [US/US]; 8660 Foxcroft Place, San Diego, CA 92129 (US).

(74) Agent: LIPSITZ, Barry, R.; Law Offices of Barry R. Lipsitz, Building No. 8, 755 Main Street, Monroe, CT 06468 (US)

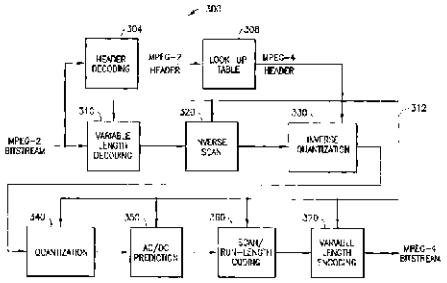
(81) Designated States (*national*): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States (*regional*): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:
with international search report
before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments

[Continued on next page]

(54) Title: VIDEO SIZE CONVERSION AND TRANSCODING FROM MPEG-2 TO MPEG-4



(57) Abstract: A transcoder architecture that provides the lowest possible complexity with a small error, e.g., for converting an MPEG-2 bitstream into an MPEG-4 bitstream. The transcoder reads header information (304) from an input bitstream and provides a corresponding header in the new format for the output bitstream. In one embodiment (Fig. 3), a low complexity frame-to-frame transcoder (with B frames disabled) avoids the need for motion compensation processing. In another embodiment (Fig. 4), a transcoder architecture that minimizes drift error (with B frames enabled) is provided. In another embodiment (Fig. 5), a size transcoder (with B frames enabled) is provided, e.g., to convert a bitstream of ITU-R 601 interlaced video coding with MPEG-2 MP@ML into a simple profile MPEG-4 bitstream which contains SIF progressive video suitable for a streaming video application. For spatial downscaling of field-mode DCT blocks, vertical and horizontal downscaling techniques are combined to use sparse matrixes to reduce computations.

WO 01/095633 A3

WO 01/095633 A3

(88) Date of publication of the international search report:
22 August 2002

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

【国際調査報告】

INTERNATIONAL SEARCH REPORT

International Application No
PCT/US 01/40811

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 H04N7/26

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)
IPC 7 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, INSPEC, COMPENDEX, PAJ

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	DOGAN S ET AL: "EFFICIENT MPEG-4/H/263 VIDEO TRANSCODER FOR INTEROPERABILITY OF HETEROGENEOUS MULTIMEDIA NETWORKS" ELECTRONICS LETTERS, IEE STEVENAGE, GB, vol. 35, no. 11, 27 May 1999 (1999-05-27), pages 863-864, XP000908120 ISSN: 0013-5194	1,22
Y	paragraph 'Introcuction! figure 1	2
A	paragraph 'Methodology of transcoding!	3-5, 10-14
	--- -/--	

[X] Further documents are listed in the continuation of box C. [X] Patent family members are listed in 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

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
29 April 2002	12 06 2002

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax. (+31-70) 340-3016	Georgiou, G

Form PCT/ISA/210 (second sheet) (July 1998)

INTERNATIONAL SEARCH REPORT

International Application No
PCT/US 01/40811

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	WEE S J ET AL: "FIELD-TO-FRAME TRANSCODING WITH SPATIAL AND TEMPORAL DOWNSAMPLING" PROCEEDINGS 1999 INTERNATIONAL CONFERENCE ON IMAGE PROCESSING. ICIP'99. KOBE, JAPAN, OCT. 24 - 28, 1999, INTERNATIONAL CONFERENCE ON IMAGE PROCESSING, LOS ALAMITOS, CA: IEEE, US, vol. 4 OF 4, 24 October 1999 (1999-10-24), pages 271-275, XP000895525 ISBN: 0-7803-5468-0 paragraph '05.1!; figure 4	2
E	PATENT ABSTRACTS OF JAPAN vol. 2000, no. 24, 11 May 2001 (2001-05-11) & JP 2001 204026 A (SONY CORP), 27 July 2001 (2001-07-27) abstract & US 2001/010706 A1 (TAKAHASHI KUNIAKI ET AL) 2 August 2001 (2001-08-02) abstract; figure 11	1-5, 10-14,22
A	WEE S J ET AL: "EFFICIENT PROCESSING OF COMPRESSED VIDEO" CONFERENCE RECORD OF THE 32ND ASILOMAR CONFERENCE ON SIGNALS, SYSTEMS & COMPUTERS. PACIFIC GROVE, CA, NOV. 1 - 4, 1998, ASILOMAR CONFERENCE ON SIGNALS, SYSTEMS AND COMPUTERS, NEW YORK, NY: IEEE, US, vol. 1, 1998, pages 855-859, XP001032864 ISBN: 0-7803-5149-5 paragraph '0003!; figure 1	1-5, 10-14,22
A	WO 97 39584 A (IMEDIA CORP) 23 October 1997 (1997-10-23) page 7, line 1 -page 10, line 15	1-5, 10-14,22
A	MORRISON D G ET AL: "REDUCTION OF BIT-RATE OF COMPRESSED VIDEO WHILE IN ITS CODED FORM" INTERNATIONAL WORKSHOP ON PACKET VIDEO, XX, XX, 1 September 1994 (1994-09-01), pages D171-D174, XP002075303 the whole document	1-5, 10-14,22
P,A	EP 1 032 214 A (MATSUSHITA ELECTRIC IND CO LTD) 30 August 2000 (2000-08-30) page 5, line 34 -page 7, line 19	1-5, 10-14,22

-/--

Form PCT/ISA/210 (continuation of second sheet) (July 1998)

INTERNATIONAL SEARCH REPORT

International Application No
PCT/US 01/40811

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	SONG J ET AL: "FAST EXTRACTION OF SPATIALLY REDUCED IMAGE SEQUENCES FROM MPEG-2 COMPRESSED VIDEO" IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, IEEE INC. NEW YORK, US, vol. 9, no. 7, October 1999 (1999-10), pages 1100-1114, XP000853341 ISSN: 1051-8215	15-20,23
A	paragraph '0009!	6-9,21
A	US 5 926 573 A (NAIMPALLY SAIPRASAD V ET AL) 20 July 1999 (1999-07-20) column 14, line 45 -column 16, line 56; figures 6,7A,7B	6-9, 15-21,23
A	EP 0 794 674 A (HEWLETT PACKARD CO) 10 September 1997 (1997-09-10) page 6, line 35 -page 16, line 30	6-9, 15-21,23
A	DUGAD R ET AL: "A fast scheme for downsampling and upsampling in the DCT domain" IMAGE PROCESSING, 1999. ICIP 99. PROCEEDINGS. 1999 INTERNATIONAL CONFERENCE ON KOBE, JAPAN 24-28 OCT. 1999, PISCATAWAY, NJ, USA,IEEE. US, 24 October 1999 (1999-10-24), pages 909-913, XP010369046 ISBN: 0-7803-5467-2 cited in the application the whole document	6-9, 15-21,23
A	YIM C ET AL: "AN EFFICIENT METHOD FOR DCT-DOMAIN IMAGE RESIZING WITH MIXED FIELD/FRAME-MODE MACROBLOCKS" IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, IEEE INC. NEW YORK. US, vol. 9, no. 5, August 1999 (1999-08), pages 696-700, XP000848395 ISSN: 1051-8215 cited in the application the whole document	6-9, 15-21,23

-/--

Form PCT/ISA/210 (continuation of second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

International Application No
PCT/US 01/40811

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	SHEN B ET AL: "ADAPTIVE MOTION-VECTOR RESAMPLING FOR COMPRESSED VIDEO DOWNSCALING" IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, IEEE INC. NEW YORK, US, vol. 9, no. 6, September 1999 (1999-09), pages 929-936, XP000848857 ISSN: 1051-8215 cited in the application the whole document	6-9, 15-21,23

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US 01/40811

Box I Observations where certain claims were found unsearchable (Continuation of item 1 of first sheet)

This international Search Report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. ☐ Claims Nos.:
because they relate to parts of the international Application that do not comply with the prescribed requirements to such an extent that no meaningful International Search can be carried out, specifically:

3. ☐ Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box II Observations where unity of invention is lacking (Continuation of item 2 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

see additional sheet

As a result of the prior review under R. 40.2(e) PCT,
no additional fees are to be refunded.

1. [X] As all required additional search fees were timely paid by the applicant, this international Search Report covers all searchable claims.

2. ☐ As all searchable claims could be searched without effort justifying an additional fee, this Authority did not invite payment of any additional fee.

3. ☐ As only some of the required additional search fees were timely paid by the applicant, this International Search Report covers only those claims for which fees were paid, specifically claims Nos.:

4. ☐ No required additional search fees were timely paid by the applicant. Consequently, this International Search Report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

Remark on Protest

[X] The additional search fees were accompanied by the applicants protest.

☐ No protest accompanied the payment of additional search fees.

Form PCT/ISA/210 (continuation of first sheet (1)) (July 1998)

International Application No. PCT/US 01 A0811

FURTHER INFORMATION CONTINUED FROM PCT/ISA/ 210

This International Searching Authority found multiple (groups of) inventions in this international application, as follows:

1. Claims: 1-5,10-14,22

Format and bit rate transcoder between two video formats by partially decompressing the bitstream.

1.1. Claims: 1-5,22
Format transcoder between MPEG-2 and MPEG-4 by recovering and transforming the header and partially decompressing the bitstream. The header transformation is performed using a lock up table. The partially decompressed data include motion vectors and DCT coefficients.

1.2. Claims: 10-14
Bit rate transcoder with partial decoding.

2. Claims: 6-9, 15-21, 23

A method for downscaling video data in the DCT domain.

Please note that all inventions mentioned under item 1, although not necessarily linked by a common inventive concept, could be searched without effort justifying an additional fee.

INTERNATIONAL SEARCH REPORT

[redacted]tional Application No
PCT/US 01/40811

Patent document cited in search report		Publication date	Patent family member(s)		Publication date
JP 2001204026	A	27-07-2001	US	2001010706 A1	02-08-2001
WO 9739584	A	23-10-1997	AU	2453897 A	07-11-1997
			CA	2249606 A1	23-10-1997
			EP	0893027 A1	27-01-1999
			JP	2000508855 T	11-07-2000
			WO	9739584 A1	23-10-1997
			US	2001055336 A1	27-12-2001
EP 1032214	A	30-08-2000	CN	1264988 A	30-08-2000
			EP	1032214 A2	30-08-2000
			JP	2000312363 A	07-11-2000
US 5926573	A	20-07-1999	US	5737019 A	07-04-1998
			CN	1182331 A	20-05-1998
			EP	0786902 A1	30-07-1997
			JP	9233316 A	05-09-1997
EP 0794674	A	10-09-1997	US	5708732 A	13-01-1998
			EP	0794674 A2	10-09-1997
			EP	0798927 A2	01 10-1997
			JP	9331532 A	22-12-1997

フロントページの続き

(81)指定国 AP(GH,GM,KE,LS,MW,MZ,SD,SL,SZ,TZ,UG,ZW),EA(AM,AZ,BY,KG,KZ,MD,RU,TJ,TM),EP(AT,BE,CH,CY,DE,DK,ES,FI,FR,GB,GR,IE,IT,LU,MC,NL,PT,SE,TR),OA(BF,BJ,CF,CG,CI,CM,GA,GN,GW,ML,MR,NE,SN,TD,TG),AE,AG,AL,AM,AT,AU,AZ,BA,BB,BG,BR,BY,BZ,CA,CH,CN,CO,CR,CU,CZ,DE,DK,DM,DZ,EE,ES,FI,GB,GD,GE,GH,GM,HR,HU,ID,IL,IN,IS,JP,KE,KG,KP,KR,KZ,LC,LK,LR,LS,LT,LU,LV,MA,MD,MG,MK,MN,MW,MX,MZ,NO,NZ,PL,PT,RO,RU,SD,SE,SG,SI,SK,SL,TJ,TM,TR,TT,TZ,UA,UG,US,UZ,VN,YU,ZA,ZW

(72)発明者 パヌソポン，クリット
アメリカ合衆国カリフォルニア州９２１２６，サンディエゴ，カーロール・キャニオン・ロードナンバー・エフ４，９６５６
(72)発明者 チェン，キュエミン
アメリカ合衆国カリフォルニア州９２１２９，サンディエゴ，フォックスクロフト・プレース８５６０

Ｆターム(参考) 5C059 KK38 KK39 KK40 KK41 LA05 LB04 LB05 MA00 MA05 MA23
MC23 MC33 MC35 NN21 PP05 PP06 PP07 PP16 RB09 RC14
SS08 SS09 UA02 UA12 UA38