

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4216956号  
(P4216956)

(45) 発行日 平成21年1月28日 (2009. 1. 28)

(24) 登録日 平成20年11月14日 (2008. 11. 14)

(51) Int. Cl.

F I

**H03M 7/30 (2006.01)**  
**H04N 1/40 (2006.01)**  
**H04N 1/41 (2006.01)**  
**H04N 7/26 (2006.01)**

H03M 7/30 B  
H04N 1/40 F  
H04N 1/41 Z  
H04N 7/13 Z

請求項の数 6 (全 8 頁)

(21) 出願番号 特願平11-180851  
(22) 出願日 平成11年6月25日 (1999. 6. 25)  
(65) 公開番号 特開2000-59230 (P2000-59230A)  
(43) 公開日 平成12年2月25日 (2000. 2. 25)  
審査請求日 平成18年6月22日 (2006. 6. 22)  
(31) 優先権主張番号 106732  
(32) 優先日 平成10年6月29日 (1998. 6. 29)  
(33) 優先権主張国 米国 (US)

(73) 特許権者 596170170  
ゼロックス コーポレーション  
XEROX CORPORATION  
アメリカ合衆国 コネチカット州 スタン  
フォード、ロング・リッジ・ロード 80  
0  
(74) 代理人 100082164  
弁理士 小堀 益  
(74) 代理人 100105577  
弁理士 堤 隆人  
(72) 発明者 ウォク・エイチ・ンギュエン  
アメリカ合衆国 カリフォルニア州 90  
807 ロングビーチ カリフォルニア  
ベニュー 3831

最終頁に続く

(54) 【発明の名称】 画像の境界のH V Q圧縮方法

(57) 【特許請求の範囲】

【請求項 1】

画素のブロックを圧縮する方法であって、

画素のブロックに編成されたオリジナルの画素を生成する段階と、

それぞれの画素のブロックに関連付けられる画像形式信号を生成する段階であって、前記画像形式信号が、異なる画像形式データタイプのグループから選択された一つのタイプが画素のブロック内に存在するか否か、または、画素のブロックが二つの画像形式データタイプ間の境界を含んでいるか否かを示す段階と、

境界が検出された場合、その境界を形成する二つの画像形式データタイプを判定する段階と、

前記画像形式信号に応答し、二つの圧縮器のうち、前記判定された二つの画像形式データタイプ間の境界を含む画素のブロックの圧縮のために構成された一つの圧縮器を選択して使用する段階と、

前記選択された圧縮器によって画素のブロックを圧縮することにより圧縮ブロック信号を形成する段階とを含み、

圧縮器のうち少なくとも一つを階層的ベクトル量子化に使用し、さらに、前記画像形式データタイプのグループが、ハーフトーン画像データ、コントーン画像データ、テキスト画像データ及びグラフィック画像データを含む、画素のブロックを圧縮する方法。

【請求項 2】

前記画像形式データタイプの一つが、ハーフトーン画像データより成る請求項 1 に記載

の方法。

【請求項 3】

前記画像形式データタイプの一つが、コントーン画像データより成る請求項 1 に記載の方法。

【請求項 4】

前記画像形式データタイプの一つが、テキスト画像データより成る請求項 1 に記載の方法。

【請求項 5】

前記画像形式データタイプの一つが、グラフィック画像データより成る請求項 1 に記載の方法。

10

【請求項 6】

誤り信号を生成する段階をさらに含む請求項 1 に記載の方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

圧縮すべき画素のブロックが 2 種類の画像データのうちのいずれであるか、あるいは 2 種類のデータ間の境界を含んでいるかを判定することによって、ブロックを圧縮するために使用される損失圧縮器が第 1 または第 2 の種類のデータ用に、または双方のデータの過渡状態用に最適化されるようにするプロセスである。

【0002】

20

【従来の技術】

データを伝送、または記憶しなければならない場合、伝送時間を短縮し、または記憶量を節減するために、通常は先ずデータが圧縮される。これは 8 ビット画素の 4 つの色分解からなることができる画像データの場合に特に当てはまる。

【0003】

圧縮方法の 1 つは、データ・ブロックを多数のステップで単一のコード語へと縮小することができる“階層的ベクトル量子化(hierarchical vector quantization)” (HVQ) である。この方法は本明細書に参考文献として引用されている米国特許明細書第 5, 602, 589 号に記載されている。この手順は上記特許の図 1 a に記載の数字の例を用いることで最も明解に説明することができる。画素当たり 8 ビットの、 $2 \times 4$  画素のブロックを 1 つの 9 ビット数に圧縮する場合を想定してみる。先ず、段階 1 の参照用テーブル (LUT) に各コード語が 2 画素ビット・パターンと関連している 9 ビット・コード語を含む各々対の画素が入力される。テーブルは、入力された画素ビットが正確に整合しない場合は、代わりに整合に最も近いコード語が出力されるように設定されている。このステップ 1 では、各々のコード語が入力された 2 つの画素のビット・パターンとの整合に最も近い 4 つのコード語が最終的に出力される。正確な整合は不可能である場合が多いので、生来この圧縮にはある程度の損失が伴う。第 1 段階の出力は 4 つの 9 ビットコード語である。

30

【0004】

この圧縮量が不十分である場合は、次に、各々が  $1 \times 2$  画素のブロックのビット・パターンを表す上記の 4 つのコード語が段階 2 の 2 つのテーブルに入力され、その結果、各々が  $2 \times 2$  画素のブロックのビット・パターンを表す総計 2 つの 9 ビット出力コード語が生ずる。それでもなお圧縮が不十分である場合は、これらの 2 つのコード語が段階 3 のテーブルに入力され、 $2 \times 4$  画素のブロックに関連する最終的な出力語が生成される。ここで、この方法が複数の段階が利用され、入力画素が互いに方向性の関係を有しているのでベクトルを含んでいる点で階層的な方法であり、かつ任意の圧縮度を達成するために多数のレベルの処理を実施できる点で量子化された方法であることがわかる。

40

【0005】

復元するには、単に各コード語が  $256K \times 64$  ビットの LUT に入力され、これが 8 画素のブロック全体のビット・パターンを出力する。

50

## 【 0 0 0 6 】

## 【 発明が解決しようとする課題 】

H V Qを利用するプリント・システムでは、データが画素のブロック内の損失圧縮器に供給され、各ブロックはテキスト、コントーン ( contone)、ハーフトーン等である。受け取られるデータの種類に応じて圧縮器自体を最適に構成できるようにするため、データの前端部は圧縮器に対して、異なる種類のデータ間の境界を判定するマスクと、各マスク内に入るデータようにどのような構成を用いればよいかを指定するプリントのヒントとを供給できる。このシステムでは、マスクのエッジをブロックのエッジと位置合わせしてければならない。

## 【 0 0 0 7 】

## 【 課題を解決するための手段 】

このシステムは過渡ブロックと呼ぶことができるブロック内に境界があるか否かを判定するシステムによって改良することができる。例えば、テキスト文字とコントーン ( 逆トーン ) の画像との境界がブロックを通る場合は、プリントのヒントが圧縮器に送られ、過渡ブロック用に最適化されたテーブルの値をそのブロック用に選択することができる。それによって、圧縮器がより少ない損失で動作できるので、システムの効率が高まるであろう。

## 【 0 0 0 8 】

## 【 発明の実施の形態 】

基本的H V Qシステムは図 1 に示したように誤りチャネルを加えることによって改良することができる。上方のチャネルでは、グレースケール・バイト・マップ 1 6 が通常の方法でH V Qデコーダ 1 0 に入力され、出力は損失なくL Zエンコーダ 1 1 で圧縮され、その結果が通常は 8 - 1 0 ビット語の形式でデコーダに送られる。

## 【 0 0 0 9 】

加えて、H V Qエンコーダの出力は走査線様式でデコーダ 1 2 に送られ、このデコーダが、損失エンコーダで誤りが誘発されることがあるので、オリジナルのバイト・マップとは異なることがあるバイト・マップのバージョンを生成する。次に 2 つのバイト・マップが画素毎に減算されて ( 1 3 )、誤り項が生成される。これが出力コード語に付加されると、オリジナルのバイト・マップが生成される。この減算は排他的O Rを利用して実行可能であり、この方が簡単かつ迅速であり、符号ビットを必要としない。各々が符号付き数量の 8 ビット幅またはそれ未満であるこれらの誤り項は次にL Zエンコーダ 1 5 内で圧縮され、オリジナルの出力と並行してデコーダに送られる。誤り項が大きい程、圧縮率は低くなる。実際には、少量の誤りは視覚的に知覚されない。圧縮率の劣化を制限するため、誤り項を量子化器 1 4 内で少数の例えば 3 つの最上位ビットに制限することができる。通常は、画素当たりの誤りの量は僅かなM S B ( 最上位ビット ) で目立つほどには大きくなく、この場合は誤り項は全くない。

## 【 0 0 1 0 】

量子化された誤りデコーダが図 2 に示されている。圧縮されたコード語はデコーダ 2 1 内でL Z復号され、デコーダ 2 2 内でL Z復号化されて、加算器 2 3 の 1 つの項が生成される。圧縮され、量子化された誤りはデコーダ 2 4 内でL Z復号され、加算器 2 3 の別の項として入力される。この加算器の出力は出力ビデオである。加算器 2 3 は符号ビットおよび 7 つのビットを加算する加算器でもよく、またはエンコーダで誤り語を生成するために用いられた場合は排他的O Rでもよい。

## 【 0 0 1 1 】

図 3 は単一コード語 3 1 が最上位および最下位部分 3 2、3 3 へと分割された場合の、かつ最下位ビットだけが圧縮される構成を示している。この場合は、ビット 0 から 4 が損失圧縮器 3 5 へと送出され、一方ビット 5 から 7 は送出されない。次に双方とも損失なしのL Z圧縮を用いて圧縮され、図 4 に示したデコーダへと出力される。この場合も、双方のチャネルがL Z復元され、一方、L S B ( 最下位ビット ) だけがH V Q復号される。次に、結果として生じた 2 つの部分が排他的論理和され ( 4 4 )、復号参照用テーブルに入力

10

20

30

40

50

される。

【 0 0 1 2 】

図 5 は単一画素が最下位セグメントと最上位セグメントとに分離され、かつ各セグメント毎に別個の異なる圧縮プロセスが実行され、最下位ビットの方がより高い圧縮率で圧縮されるシステムである。オリジナルの画素は最上位ビット 5 2 と最下位ビット 5 3 とに分離される。その結果、最も重要である最上位ビットはより低い圧縮率で圧縮され、一方、最下位ビットの圧縮率は高い。プログラム可能な参照用テーブルを利用して、入力された画素を図示した 3 - 5 分割以外の任意の 2 つのセグメントへと分割することもできよう。

【 0 0 1 3 】

図 6 は図 5 のエンコーダ用のデコーダである。図 5 の圧縮された 2 つの出力は L Z デコーダ 6 1、6 2 に入力され、かつ復号されて H V Q 6 3、6 4 に入力される。次に、双方の出力が排他的 O R ゲート 6 5 に入力されて、画素全体が組立てられる。勿論、エンコーダがそれ以外の何らかのエンコーダの組合せを使用した場合は、デコーダは同じ復号形式を用いるであろう。すなわち、より一般的には、ストリング内のデータ語をより上位の、または下位のビットに分割して 2 つの平行ストリングを作成することができ、次に下位のビットにはより高い圧縮率での圧縮が行われる、2 つの圧縮方法のいずれかを用いて圧縮することができる。

【 0 0 1 4 】

H V Q 圧縮は図 7 に示すように画像の回転および鏡像写像に容易に適応する。この図には、高さが 4 画素で幅が 1 6 画素であり、時計回り方向に 9 0 ° 回転され、鏡像を写像しなければならないオリジナルの画像の例を用いたプロセスが示されている。

【 0 0 1 5 】

ステップ 1 は 8 つの 8 画素セグメントを 8 つのコード語 C w1 から C w8 へと縮小する通常の圧縮プロセスである。ステップ 2 は、コード語を回転させ、鏡像形成された順に再構成するステップである。このハードウェアは、図示のように 2 番目の語イン、例えば C w2 が 3 番目の語アウトに接続される配線の形式でよい。ステップ 2 は各組の配線が異なる回転をもたらす数組の配線を有することができる。最後に、復号ステップ 3 は参照用テーブルを使用して、適切な向きにある各セグメント毎の画素パターンを作成する。ここでまた数個のテーブルを使用して各種の向きを生成することができる。

【 0 0 1 6 】

プリンタに対してデータの最良のプリント方法を指示するために、ユーザーによって供給されるオリジナルのデータにオリジナルのページ記述言語でプリントのためのヒントを組込んでもよい。例えば、ヒント語は 2 ビットの長さでよく、後続のデータがテキスト、コントーン(contone)、グラフィック等である 4 つの可能性の 1 つを指示する。例えば、プリンタがコンピュータで作成したグラフィックとして発信されたデータを受信した場合は、プリンタはオリジナルのデータが写真の形式でスクリーニングされている場合とは異なるハーフトーンのスクリーンを使用できる。

【 0 0 1 7 】

プリントのヒントは図 5 に示すようにどの H V Q チャンネルに付加してもよい。H V Q エンコーダ 5 4 に入力される各々 4 画素・ブロック毎に、2 つの 8 ビット・バイトに含まれている長さ 9 ビットの 1 つのコード語が作成され、従って、最初のバイトには最初の 8 ビットが含まれ、2 番目のバイトには最後のビットが含まれているものと想定してみる。次に 2 ビットのヒントが付加される。この時点で、各コード語とヒントをプラスした 1 1 ビットがなお 2 バイト内に含まれている。L Z エンコーダは恐らくは長さが 2 5 6 バイトのバイト・ストリングを探索し、最新の同一パターンの整合の位置とサイズとを判定する。このストリング内でヒントが一度または二度変化する範囲までは、ヒントがない場合に生ずる圧縮量と比較して圧縮量の減少は僅かである。しかし、この期間中にヒントが変化しない限りは、圧縮量は全く減少しない。その結果、プリントのヒントは H V Q 圧縮器の後に、しかし損失なしの圧縮器の前に供給することができ、そうしても圧縮率にはほとんど影響しない。出力側で、コード語が L Z 復元された後で、しかし復号される前に、ヒントを

10

20

30

40

50

事後に利用するために抽出することができる。

【 0 0 1 8 】

実際の入力データ・パターンと整合する最良のチャンスを有するコード語と出力データ・パターンを選択することによって、H V Q圧縮器の損失を更に最小限にすることができる。テキスト画素が4×2画素の群で符号化され、かつ1行の4つの入力画素が黒、ダークグレー、ライトグレーおよび白であり、また入力ビデオがスキャン・イン・テキストであった場合、オリジナルのデータは黒の文字と白の空白との境界であった公算が強く、従って出力画素パターンは黒、黒、白、白であろう。これに対して、オリジナルの入力データがスキャン・インされたコンピュータ作成グラフィックであった場合は、4つの画素は黒から白に密度がスムーズに低下している公算が強い。デコーダの参照用テーブル内のエンコードコード語とパターンの実際の判定は統計的分析によって決定される。テキストおよびグラフィック文書の代表的な群がテスト・プログラムを通され、各タイプ毎に最良の値が生成される。

10

【 0 0 1 9 】

境界が画素の入力ブロックを通過する場合に混乱が生じ、この場合はブロック全体にテキスト値もグラフィック値も用いることができない。その解決方法は、このような境界状態のために特別に生成された第3組のコード語とパターンとを供給することである。この場合、テキストとグラフィックの双方を含む文書セットが分析されて、1組のパターンが作成され、これは境界の混合状態がブロック内にあるものと判定された場合に利用されよう。

20

【 0 0 2 0 】

境界状態はプリントのヒントを観察することによって検出される。例えば、方形のスキャン・イン画像は標準的にはテキストのページにそのx, y座標によって位置決めされる。ラスタ出力スキャナ・ビームが座標内にある場合は、プリントのヒントはプリンタに対して、どのコード語、参照用テーブルの項目、およびハーフトーン・スクリーンを使用すべきかを指示する。ヒントが例えばブロック内で画像からテキストに変化した場合は、エンコードはブロック内に境界が存在することを認識する。

【 0 0 2 1 】

いずれの場合もブロックのサイズは保持されなければならない。図8はテキストからコントーンへの移行の例を示している。H V Qエンコードでは異なる種類のデータ用に異なるブロック・サイズを用いることができるので、ここではテキスト用のブロック・サイズはエッジをより細密にできるように2×2画素として示されており、一方、コントーン用のブロック・サイズは圧縮率を高めることができるように4×2画素として示されている。

30

【 0 0 2 2 】

境界が2×2画素のブロック内にある場合は、このブロックは境界値を用いて符号化され、かつ復号される。加えて、画素81のような境界の右にあるいずれかのコントーン画素も必要ならば境界画素として処理されるので、境界線の右にある残りの全ての画素は4×2画素のブロック内にくる。同様にして、テキストおよびコントーンの4×2画素のブロックの境界を示す図9では、2×2画素の境界ブロックが一对で用いられるので、残りのブロックすべては4×2画素のブロックとなる。

40

【図面の簡単な説明】

【図1】 並列の誤りチャンネルを有するH V Qエンコードである。

【図2】 図1の構成用のデコードである。

【図3】 最上位ビットと最下位ビットとに分割された画素を符号化するための1チャンネル構成図である。

【図4】 最上位ビットと最下位ビットとに分割された画素を復号するための1チャンネル構成図である。

【図5】 最上位ビットと最下位ビットとに分割された画素を符号化するための2チャンネル構成図である。

【図6】 図5のエンコード用のデコードである。

50

【図 7】 H V Q 圧縮を利用して画像を回転写像、もしくは鏡像写像する 3 つのステップを示している。

【図 8】 サイズが異なるブロック間の境界を示している。

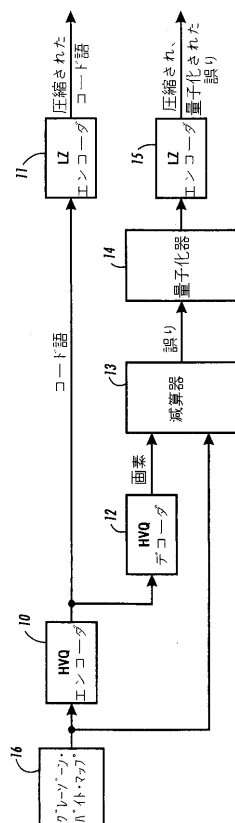
【図 9】 サイズが同じブロック間の境界を示している。

【符号の説明】

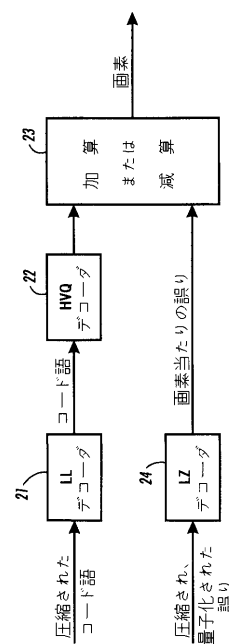
1 0 H V Q デコーダ、1 1 L Z エンコーダ、1 2 デコーダ、1 3 減算器、1 6  
グレースケール・バイト・マップ、2 1 デコーダ、2 2 デコーダ、2 3 加算器、2  
4 デコーダ、3 1 単一コード語、3 2 最上位部分、3 3 最下位部分、3 4 , 3 6  
L Z エンコーダ、3 5 圧縮器、4 1 , 4 2 L Z エンコーダ、4 3 H V Q デコーダ  
、4 4 排他的 O R ゲート、5 2 最上位ビット、5 3 最下位ビット、5 4 H V Q エ  
ンコーダ、6 1 L Z デコーダ、6 2 L Z デコーダ、6 3 H V Q、6 4 H V Q、6  
5 排他的 O R ゲート

10

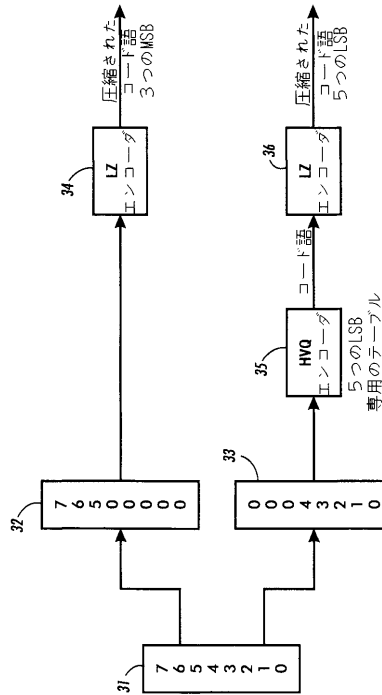
【図 1】



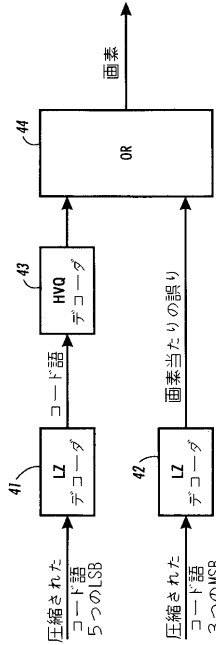
【図 2】



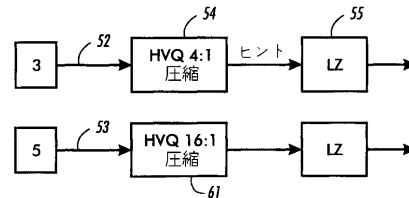
【図 3】



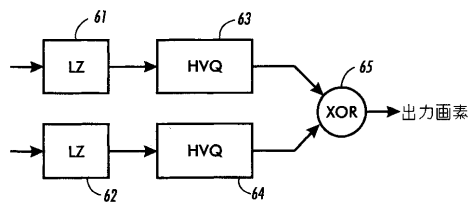
【図 4】



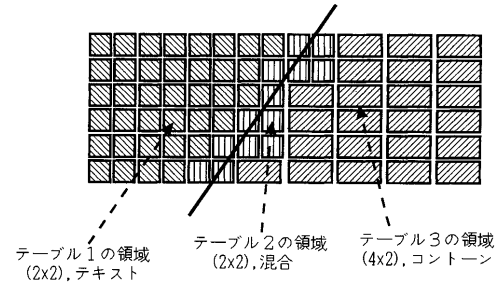
【図 5】



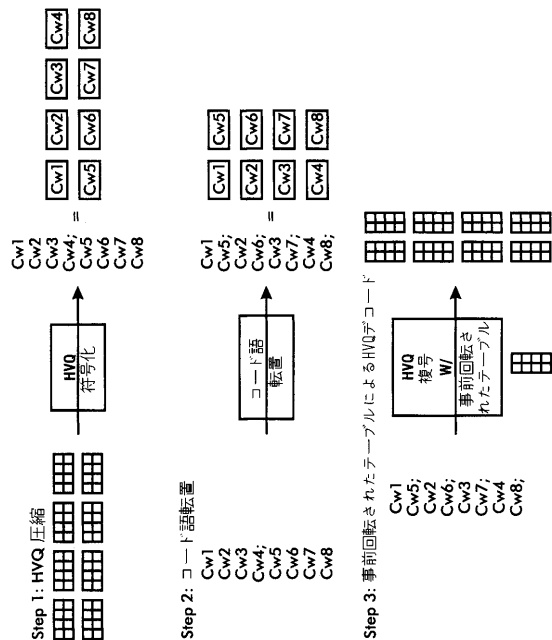
【図 6】



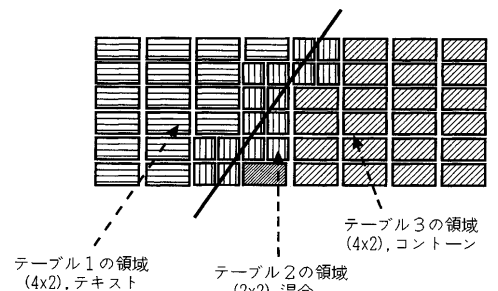
【図 8】



【図 7】



【図 9】



---

フロントページの続き

- (72)発明者 キエン・ティー・ンギュイエン  
アメリカ合衆国 カリフォルニア州 9 0 5 0 1 トーランス エルドラドストリート 2 1 7 2
- (72)発明者 エイブラハム・イー・クラブロス  
アメリカ合衆国 カリフォルニア州 9 0 2 3 0 カルバーシティー オーバーランドアベニュー  
4 5 5 0
- (72)発明者 ウエイリ・リン  
アメリカ合衆国 カリフォルニア州 9 2 6 1 2 アーバイン ヤングコート 2 0

審査官 渡辺 未央子

- (56)参考文献 特開平 0 7 - 0 3 8 7 6 1 ( J P , A )  
特開平 0 9 - 0 9 3 5 7 6 ( J P , A )

(58)調査した分野(Int.Cl. , D B 名)

H03M 7/30  
H04N 1/40  
H04N 1/41  
H04N 7/24