

(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2004-532548
(P2004-532548A)

(43) 公表日 平成16年10月21日(2004.10.21)

(51) Int.CI.⁷HO4N 5/92
HO4N 1/41
HO4N 5/91

F 1

HO4N 5/92
HO4N 1/41
HO4N 5/91H
B
R

テーマコード(参考)

5C053
5C078

審査請求 未請求 予備審査請求 有 (全 69 頁)

(21) 出願番号 特願2002-574576 (P2002-574576)
 (86) (22) 出願日 平成14年3月19日 (2002.3.19)
 (85) 翻訳文提出日 平成15年9月18日 (2003.9.18)
 (86) 國際出願番号 PCT/US2002/008783
 (87) 國際公開番号 WO2002/075640
 (87) 國際公開日 平成14年9月26日 (2002.9.26)
 (31) 優先権主張番号 60/277,609
 (32) 優先日 平成13年3月19日 (2001.3.19)
 (33) 優先権主張国 米国(US)
 (31) 優先権主張番号 60/365,203
 (32) 優先日 平成14年3月15日 (2002.3.15)
 (33) 優先権主張国 米国(US)

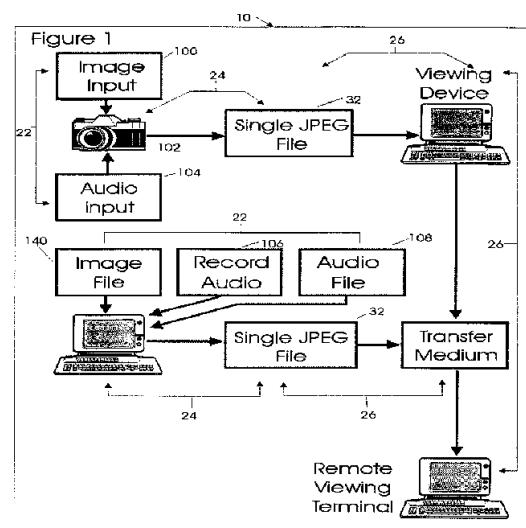
(71) 出願人 503340656
 サウンドピックス・インク
 アメリカ合衆国、ネバダ州 89451、
 インクライン・ビレッジ、ブリジャー・コ
 ート 689
 (74) 代理人 100071010
 弁理士 山崎 行造
 (74) 代理人 100104086
 弁理士 岩橋 越夫
 (74) 代理人 100121762
 弁理士 杉山 直人
 (74) 代理人 100126767
 弁理士 白銀 博
 (74) 代理人 100122839
 弁理士 星 貴子

最終頁に続く

(54) 【発明の名称】 JPEGファイル内のデータ保存システム及び方法

(57) 【要約】

本発明は、画像データと他のデータとの交信のためのシステムを具備する。本システムは、JPEGデータを有するJPEGファイル、少なくとも1つのJPEGでないデータとコンピュータとを具備する。コンピュータは、あらかじめ定められたシーケンシャルバイト数のJPEGでないデータを読み込むことがプログラムされている。またコンピュータは、JPEGマーカーを含んでおり、ローケーターバイトを決定するシーケンシャルバイトにおいて、各バイトの位置を決定するようにプログラムされている。ここで、ローケーターバイトはJPEGマーカーを含むシーケンシャルバイトにおけるすべてのバイトの位置を表示することができる。コンピュータはさらに、ローケーターバイトをJPEGファイルに書き込み、エンコードされたバイトをJPEGファイルに書き込むことがプログラムされている。コンピュータは、シーケンシャルバイトの各バイトに対して、もしそのバイトがJPEGマーカーを含んでいなければ、シーケンシャルバイトにおいて、このようなバイトが発生した順に、各バイトを好ましくは0と書かれたビットに置き換えるようなやり方で、JPEGファイル内に



【特許請求の範囲】**【請求項 1】**

JPEGファイル内にJPEGでないデータを保存する方法であって、

(A) JPEGでないデータにおいて、あらかじめ定められた数のシーケンシャルバイトを読み込むステップと、

(B) 当該シーケンシャルバイトにおいて、JPEGマークーを含む各バイトの位置を決定するステップと、

(C) JPEGマークーを含むシーケンシャルバイトの各バイトの位置を表示することができるローケーターバイトを決定するステップと、

(D) エンコードされたバイトを決定するステップと、

を含み、

エンコードされたバイトには、

(a) ローケーターバイトと、

(b) JPEGマークーを含まないシーケンシャルバイトの各バイトと、

(c) JPEGマークーを含むシーケンシャルバイトの各バイトに対するプレースホールダーバイトと、

を具備する方法。

【請求項 2】

エンコードされたバイトをあらかじめ定められた順番に書きこむステップを含む、請求項1に記載の方法。

【請求項 3】

ローケーターバイトが最も目立つ位置に書きこまれる、請求項2に記載の方法。

【請求項 4】

シーケンシャルバイト内でその発生順にシーケンシャルバイトが書きこまれ、プレースホールダーバイトが置き換えたバイトの順にプレースホールダーバイトが書きこまれる、請求項2に記載の方法。

【請求項 5】

エンコードされたバイトがJPEGファイル中に書きこまれる、請求項2に記載の方法。

【請求項 6】

JPEGファイル中のエンコードされたバイトの前にアプリケーションデータマークーを書きこむステップをさらに含む、請求項5に記載の方法。

【請求項 7】

あらかじめ定められた数のシーケンシャルバイトの数は7である、請求項1に記載の方法。

【請求項 8】

JPEGマークーは0×FFを含む、請求項1に記載の方法。

【請求項 9】

プレースホールダーバイトには0×00を含む、請求項1に記載の方法。

【請求項 10】

プレースホールダーバイトは0×00から0×7Fの間である、請求項1に記載の方法。

【請求項 11】

0×00のローケーターバイトは、シーケンシャルバイトがどれもJPEGマークーを含まないことを示す、請求項1に記載の方法。

【請求項 12】

JPEGでないデータには、音声データを含む、請求項1に記載の方法。

【請求項 13】

請求項1において、さらに、

(A) JPEGファイル中にアプリケーションデータマークーを置くステップと、

(B) JPEGファイル中の当該アプリケーションデータマークーの後にエンコードされたバイトを書きこむステップと、

10

20

30

40

50

を含む、請求項 1 に記載の方法。

【請求項 14】

請求項 1 において、さらに、

(A) 請求項 1 における A から D までのステップをすべての JPEG でないデータについて繰り返すステップと、

(B) エンコードされたバイトの総数を決定するステップと、

(C) もしエンコードされたバイトの総数が JPEG フィールド長を超えていた場合は、

(a) エンコードされたファイルをセグメントに分割し、各セグメントにおけるバイト数を最大 JPEG フィールド長より短くするステップと、

(b) セグメントを JPEG ファイル中に書きこむステップと、

を含む、請求項 1 に記載の方法。

【請求項 15】

エンコードされたバイトと共に JPEG ファイルを記録するステップをさらに含む、請求項 1 3 に記載の方法。

【請求項 16】

ローケーターバイト及びエンコードされたバイトと共に JPEG ファイルを伝送するステップをさらに含む、請求項 1 3 に記載の方法。

【請求項 17】

請求項 1 3 において、さらに、

(A) JPEG ファイル中のエンコードされたバイトを読み込むステップと、

(B) ローケーターバイトを決定するステップと、

(C) ローケーターバイトからエンコードされたバイト中のプレースホールダーバイトの位置を決定するステップと、

(D) 各プレースホールダーバイトを JPEG マーカーを含んだバイトに変換し、それによりエンコードされたバイトをデコードしシーケンシャルバイトを得るステップと、

を含む、請求項 1 3 に記載の方法。

【請求項 18】

アプリケーションプログラム中に、シーケンシャルバイトを使うステップをさらに含む、請求項 1 7 に記載の方法。

【請求項 19】

シーケンシャルバイトを記録するステップをさらに含む、請求項 1 7 に記載の方法。

【請求項 20】

JPEG ファイルにコントロールデータを書きこむステップをさらに含み、当該コントロールデータは JPEG でないデータに関連したものである、請求項 1 3 に記載の方法。

【請求項 21】

画像データ及び他のデータと交信を行うシステムであって、

(A) JPEG データを含有する JPEG ファイルと、

(B) JPEG でないデータと、

(C) コンピュータと、

を備え、

当該コンピュータは、

(a) JPEG でないデータにおいて、あらかじめ定められた数のシーケンシャルバイトを読み込むステップと、

(b) 当該シーケンシャルバイトにおいて、JPEG マーカーを含む各バイトの位置を決定するステップと、

(c) JPEG マーカーを含むシーケンシャルバイトの各バイトの位置を表示することができるローケーターバイトを決定するステップと、

(d) エンコードされたバイトを決定するステップと、

ここで、エンコードされたバイトには、

(i) ローケーターバイトと、

10

20

30

40

50

(ii) JPEGマークを含まないシーケンシャルバイトの各バイトと、
(iii) JPEGマークを含むシーケンシャルバイトの各バイトに対するプレースホールダーバイトと、
を備え、
(e) JPEGファイル中にエンコードされたバイトを書きこみ、エンコードされたバイトはあらかじめ定められた順番に書きこむステップと、
を含むコンピュータである、
画像データ及び他のデータと交信を行うシステム。

【請求項 2 2】

コンピュータと交信する伝送装置であって、エンコードされたバイトと共にJPEGファイルを送信するよう構成された伝送装置をさらに備える請求項 2 1 に記載のシステム。 10

【請求項 2 3】

コンピュータと交信する画像データ入力装置であって、JPEGデータを提供するよう構成された画像データ入力装置をさらに備える請求項 2 1 に記載のシステム。

【請求項 2 4】

前記画像データ入力装置にはディジタルカメラを含む、請求項 2 3 に記載のシステム。

【請求項 2 5】

コンピュータと交信するJPEGでないデータの入力装置であって、JPEGでないデータを提供するよう構成されたJPEGでないデータの入力装置をさらに備える請求項 2 1 に記載のシステム。 20

【請求項 2 6】

前記JPEGでないデータの入力装置には音声記録装置を含む、請求項 2 1 に記載のシステム。

【請求項 2 7】

前記コンピュータはアプリケーションデータマークの後のJPEGファイル中にエンコードされたファイルを書きこむようさらにプログラムされている、請求項 2 1 に記載のシステム。

【請求項 2 8】

前記コンピュータはJPEGファイル中に、JPEGでないデータと関係があるコントロールデータを書きこむようさらにプログラムされている、請求項 2 1 に記載のシステム。 30

【請求項 2 9】

請求項 21 に記載のシステムであって、前記コンピュータと交信する 2 番目のコンピュータを備え、当該 2 番目のコンピュータは、

- (A) JPEGファイル中のエンコードされたバイトを読み込むステップと、
 - (B) ローケーターバイトを決定するステップと、
 - (C) ローケーターバイトからエンコードされたバイト中のプレースホールダーバイトの位置を決定するステップと、
 - (D) 各プレースホールダーバイトをJPEGマークを含んだバイトに変換し、それによりエンコードされたバイトをデコードしシーケンシャルバイトを得るステップと、
- を実行するよう構成されたコンピュータである、請求項 21 に記載のシステム。 40

【請求項 3 0】

前記コンピュータはJPEGでないデータを用いるよう構成された請求項 2 6 に記載のシステム。

【請求項 3 1】

JPEGファイル中にJPEGでないデータを保存する方法であって、

- (A) JPEGでないデータを読み込むステップと、
 - (B) JPEGでないデータのどのバイトがあらかじめ定められたJPEGマークを包含するかを決定するステップと、
 - (C) エンコードされたバイトを決定するステップと、
- を含み、 50

エンコードされたバイトは、

- (a) JPEGマークを具備しないJPEGでないデータの各バイトと、
- (b) あらかじめ定められたJPEGマークを具備するJPEGでないデータの各バイトのプレースホールダーバイトと、
- (c) 少なくとも1つのローケーターバイトであって、どのエンコードされたバイトがプレースホールダーバイトであるかを表示することができるローケーターバイトと、を具備する、JPEGファイル中にJPEGでないデータを保存する方法。

【請求項32】

あらかじめ定められた順序で、エンコードされたバイトを書きこむステップをさらに含む請求項31に記載の方法。

10

【請求項33】

ローケーターバイトが最も目立つ位置に書きこまれる、請求項32に記載の方法。

【請求項34】

エンコードされたバイトがJPEGファイル中に書きこまれる、請求項32に記載の方法。

【請求項35】

JPEGファイル中のエンコードされたバイトの前にアプリケーションデータマーカーを書きこむステップをさらに含む、請求項34に記載の方法。

【請求項36】

JPEGマークは $0 \times F F$ を含む、請求項31に記載の方法。

【請求項37】

プレースホールダーバイトは $0 \times 0 0$ を含む、請求項31に記載の方法。

20

【請求項38】

$0 \times 0 0$ のローケーターバイトは、JPEGでないデータのどのバイトもJPEGマークを含まないことを示す、請求項31に記載の方法。

【請求項39】

JPEGでないデータには音声データを含む、請求項31に記載の方法。

【請求項40】

請求項31において、さらに

- (A) JPEGファイル中にアプリケーションデータマーカーを置くステップと、
 - (B) JPEGファイル中に当該アプリケーションデータマーカーの後にエンコードされたバイトを書きこむステップと、
- を含む、請求項31に記載の方法。

30

【請求項41】

請求項31において、さらに、

- (A) エンコードされたバイトの総数を決定するステップと、
 - (B) もしエンコードされたバイトの総数がJPEGフィールド長を超えていた場合は、
 - (a) エンコードされたファイルをセグメントに分割し、各セグメントにおけるバイト数を最大JPEGフィールド長より短くするステップと、
 - (b) セグメントをJPEGファイル中に書きこむステップと、
- を含む、請求項31に記載の方法。

40

【請求項42】

エンコードされたバイトと共にJPEGファイルを記録するステップをさらに含む、請求項34に記載の方法。

【請求項43】

ローケーターバイト及びエンコードされたバイトと共にJPEGファイルを伝送するステップをさらに含む、請求項34に記載の方法。

【請求項44】

請求項34において、さらに、

- (A) JPEGファイル中のエンコードされたバイトを読み込むステップと、
- (B) ローケーターバイトを決定するステップと、

50

(C) ローケーターバイトからエンコードされたバイト中のプレースホールダーバイトの位置を決定するステップと、

(D) 各プレースホールダーバイトを、JPEGマークーを含んだバイトに変換し、それによりエンコードされたバイトをデコードしシーケンシャルバイトを得るステップと、
を含む、請求項34に記載の方法。

【請求項45】

アプリケーションプログラム中に、シーケンシャルバイトを使うステップをさらに含む、
請求項34に記載の方法。

【請求項46】

JPEGファイルにコントロールデータを書きこむステップをさらに含み、当該コントロール
データはJPEGでないデータに関連したものである、請求項34に記載の方法。 10

【発明の詳細な説明】

【0001】

関連特許

本出願は、2002年5月15日付け米国出願出願と2001年5月19日付け米国出願
、暫定的出願番号60/277,609に基づく優先権の利益を主張する。

【0002】

発明の背景

発明の分野

本発明は、JPEGファイルデータ保存システム及び方法に関する。 20

【0003】

関連技術の説明

JPEGファイル

JPEGとは、ディジタル画像データを圧縮、保存及び伝達するために用いられるファイルフォーマットの標準である。ここで使われているとおり、「JPEG」という用語は、「JPEG」及び「JPEG2000」を含むすべてのバージョン、修正版、及び公開されたフォーマットに適用される。このフォーマットは、ジョイント・フォトグラフィック・エキスパートグループにより開発され、現在最も人気があり広く使われている画像フォーマットである。JPEGの利点の一つは、一般的に使われるどんなビットマップ・フォーマットに対しても最大限に圧縮し、装置やユーザーが最小限の保存スペースを使って画像ファイルを保存することが出来るようになることである。この圧縮能力は、JPEGが、インターネットを使って画像を送信するときのフォーマットとして最も一般的に使われるようになっているJPEGの特徴の一つでもある。ユーザーは、他の多くのフォーマットより高い画像の質を保ちながら、より高速度にJPEG画像を伝送し受け取ることができる。 30

【0004】

JPEGフォーマットはファイルをセグメントに分割するマーカーを使う。マーカーは0xF
Fの値を持つ最初のバイトを具備する。JPEG・マーカーは、2つの一般的なタイプすなわち
スタンダードアローンとデータとにグループ分けすることができる。スタンダードアローン・マ
ーカーは2バイトのマーカー以外のデータは持たない。データ・マーカーはこのマーカーの直後に、データのバイト数を示す2バイトが付随する。例えば、データ・マーカーは以
下のように現れる。 40

【0005】

0xFF 0xNN 0xCH 0xCL

この例では、0xFFはJPEGセグメントまたはフィールドの始まりを示し、0xNNはフィ
ールドのタイプを表示し、そして0xCHと0xCLは合わせてフィールドの長さを示
している。このマーカーにはデータやインストラクションを付随させても良い。

【0006】

対象となる画像のデータに加えて、JPEGは、JPEGでないデータやアプリケーションデータ
をJPEGファイルの中に蓄えられるようにすることができる。このアプリケーションデータ
は、画像データを使うアプリケーション以外のアプリケーション用データを含んでいても 50

良い。例えば、画像をとらえたカメラのセッティングあるいは画像を作った者の身元を示すデータをJPEGファイルに挿入することができる。このように、JPEGの利点の一つは、対象となる画像データ以外のデータをJPEGファイル自身に保存し伝送することができる点である。

【0007】

任意のデータをJPEGファイルに組み込むことは多くの実際のユーザーに提供されている。例えば、静止画像や写真は、恐らくはその写真について説明する人の声である音声ファイルとともに、インターネットを介して伝送することができる。このことは、家族が家族のメンバーの声と写真を共有すること、不動産業者が不動産の写真をその不動産を説明する声と共に提供すること、医療専門家が診断画像を音声による診断や説明と共に提供すること、及び、その他さまざまな用途に使うことを可能にする。10

【0008】

しかし、JPEGファイルにアプリケーションデータを保存することに関連する問題の1つは、アプリケーションデータはマーカーコードを含むことが出来ないことである。例として、デコーダーがアプリケーションデータにおける $0 \times FF$ をJPEGマーカーとして読み込む場合があり、このことが修復不可能なデコードイングのプロセスの失敗をもたらす。もしJPEGファイル中のアプリケーションデータがマーカーコードを含んでいたら、そのマーカーはJPEGデコーダーを壊してしまうので、そのファイルは使用できなくなる。

【0009】

この問題の一つの解決手段は、アプリケーションデータからすべての $0 \times FF$ のインスタンスあるいは他のマーカーコードを取り除くことである。このことによりデータの蓄積が可能となり、JPEGデコーダーを壊す恐れなしにJPEGファイル内に送り込むことができる。しかし、もし1以上のマーカーコードがアプリケーションデータから取り除かれたなら、そのデータは、通常は大きく改変され目的のアプリケーションプログラムに使うことが出来なくなる。20

【0010】

それゆえに、JPEGファイルにデータが蓄積されているときにアプリケーションデータからマーカーコードを取り除き、アプリケーションプログラムを使用するときデータをそのオリジナルの状態に戻すための、システム及び方法のニーズが存在する。

【0011】

オーディオファイルと画像ファイルの結合
任意のデータとJPEGファイルとを結合する技術は存在する。例えば、サウンドの付加されたディジタル画像を創りたいと望むユーザーは画像ファイルに別のオーディオ、例えば、wavファイルを付け加えることができる。しかしwavファイルは、伝達または蓄積される時に、しばしば簡単に書き換えられ消去されてしまう。wavファイルに関するもう一つの問題は、保存し使用するためにマイクロソフト（登録商標）のメディアソフトウェアのような特別なソフトウェアを必要とすることである。30

【0012】

他の任意のデータとJPEGファイルとを結合する技術は、Ota他により発行された米国特許番号5,032,918、及びHashimoto他により発行された米国特許番号5,815,201により公開された技術に含まれている。ここでは、オーディオファイルと画像ファイルとは別々に記録されお互いに結合される。オータその他により公開された発明の欠点の1つは、2つの別々のファイルを保持することが必要とされる点である。これは、特に大容量のファイルを扱うとき、あるいは長時間にわたって実行されるときに、大変煩わしく結合関係を失いやすいものである。40

【0013】

McIntyre他により発行された米国特許番号6,102,505は不可視性のインクにより画像データとオーディオデータとの永久的な結合を作り出すものである。Akamine他により発行された米国特許番号5,644,557及びNozaki他により発行された欧州特許番号EP0964304A1は、各々、フィルム上の磁気による記録及びフィルム上の光ピクセルによるデータ記録を公開50

している。Akamine他及びNozaki他に関連する問題の1つは、オーディオデータのハードコピーを蓄えるために物理的にフィルムを必要とすることであり、これはディジタル画像データには十分適したものではない。

【0014】

他のオーディオデータと画像データを蓄積する技術は、Paoliniにより発行された米国特許番号5,276,866により公開された発明に含まれており、ここでは、オーディオデータはビデオストリームの中の画像フレームとして蓄えられ、そして圧縮される。Case他により発行された米国特許番号5,440,677は、関連付けて再生することを可能とするファイルを付加してCD-ROMにオーディオファイルと画像ファイルを蓄積する技術を公開している。PaoliniとCase他についての問題は、先の特許と同様、データを回復、表示あるいは再生の前に、データの送信及び受信の双方ために特別なプログラムが必要なことである。10

【0015】

つまり、画像データとアプリケーションデータの結合において本来的に備わっているファイルの保守の必要性を削減するニーズが存在している。また、インターネットのような種々の媒体を介して効率的にまた便利に伝達できるような、アプリケーションデータと画像データの永久的な結合を作り出すことのニーズも存在している。さらに、効率的にまた便利にアプリケーションデータと画像データを結合し画像データを1つのファイルにするとのニーズも存在している。アプリケーションデータと画像データの保存、回復、表示、あるいは使用のために、特別なあるいはあまり広く使われていないアプリケーションを必要とせず、アプリケーションデータを画像データとともに保存することのニーズもさらに存在している。20

【0016】

発明の概要

本発明の利点

本発明の利点の1つは、画像データの添付されたアプリケーションデータの信頼性のある保存がされることである。

【0017】

本発明の他の利点は、画像デコーダーと相いれないアプリケーションデータを画像データと共に保存できることである。

【0018】

本発明の他の利点は、画像データとアプリケーションデータの結合において本来的に必要とする保守を削減することである。30

【0019】

本発明の他の利点は、アプリケーションプログラムに既存のアプリケーションを検出させることで、画像データとアプリケーションデータを結合させ1つのファイルにするシステム及び方法を提供することである。

【0020】

本発明の他の利点は、JPEGデコーダーがデータ0×FFを含むアプリケーションデータを読み込んだとき、JPEGデコーダーの回復不可能な失敗を防止する点である。

【0021】

本発明の他の利点は、マーカーコードのインスタンスを取り除き、同時に、元のアプリケーションデータの信頼性の高い再構築を可能とする順序、保全及び識別データを具備するデータを提供することである。40

【0022】

本発明の他の利点は、アプリケーションデータと画像データの保存、回復、表示、あるいは使用のために、特別なあるいはあまり広く使われていないアプリケーションを必要とせず、アプリケーションデータを画像データとともに保存するシステムと方法を提供することである。

【0023】

本発明の他の利点は、アプリケーションデータを復活させるデコーディングメカニズムを50

提供することである。

【0024】

本発明の他の利点は、各JPEGアプリケーションレコードの最初の部分に付加的なデータレコードを挿入するエンコーディングメカニズムを提供することである。

【0025】

本発明の他の利点は、アプリケーションではない多くのデータを、少なくとも1つの画像ファイルに永久的に結合させることである。

【0026】

本発明の他の利点は、データを失うことなく、元のデータをエンコードすることである。

【0027】

本発明の他の利点は、元のデータをエンコードするあいだ、再生パラメータ、言語識別、複数トラック情報、タイミング、その他の付加的な情報を含む元のデータを保存することである。

【0028】

本発明の他の利点は、受信システムがアプリケーションデータを無視したとしても、コンピュータが画像データにアクセスできるようにしていることである。

【0029】

本発明の他の利点は、音声と共にデジタル画像を提供することである。

【0030】

本発明の他の利点は、修正したアプリケーションデータの安全性を保障するような方法で、アプリケーションを画像ファイルに保存することである。

【0031】

本発明の他の利点は、アプリケーションデータの再構築を可能とする画像ファイル内のマークを提供することである。

【0032】

本発明の他の利点は、画像ファイル中に、画像ファイルの最大フィールド長さより長いデータを保存する方法を提供することである。

【0033】

本発明の他の利点は、再生アプリケーションにより使用可能なデータを画像ファイルの中に著相する方法を提供することである。

【0034】

本発明のこれらや他の利点は、本明細書の残りの部分、請求の範囲及び要約を参照することにより実現される。

【0035】

発明の簡単な説明

本発明は、画像データと他のデータとの交信のためのシステムを具備する。本システムは、JPEGデータを有するJPEGファイル、少なくとも1つのJPEGでないデータとコンピュータとを具備する。コンピュータは、あらかじめ定められたシーケンシャルバイト数のJPEGでないデータを読み込むことがプログラムされている。またコンピュータは、JPEGマークを含んでおり、ローケーターバイトを決定するシーケンシャルバイトにおいて、各バイトの位置を決定するようにプログラムされている。ここで、ローケーターバイトはJPEGマークを含むシーケンシャルバイトにおけるすべてのバイトの位置を表示することができる。コンピュータはさらに、ローケーターバイトをJPEGファイルに書き込み、エンコードされたバイトをJPEGファイルに書き込むことがプログラムされている。コンピュータは、シーケンシャルバイトの各バイトに対して、もしそのバイトがJPEGマークを含んでいなければ、シーケンシャルバイトにおいて、このようなバイトが発生した順に、各バイトを好ましくは0と書かれたビットに置き換えるようなやり方で、JPEGファイル内にエンコードされたバイトを書きこむ。もしそのバイトがJPEGマークを含んでいる場合は、JPEGマークを含むバイトは、シーケンシャルバイトにおいてこのようなバイトが発生した順に、好ましくは1と書かれたホールダービットに置き換えられる。

【 0 0 3 6 】

以上の説明は、以下に述べる最適実施例の詳細な説明が良くわかり、本発明の技術に対する寄与度がより理解できるように、本発明の重要な特徴を広く述べたものである。本発明は、以下に述べるようにまた特許請求の範囲の対象となる付加的な特徴が有ることは言うまでもない。この点について、本発明の少なくとも1つの最適実施形態の詳細説明にあたって、本発明は以下の説明に述べられ、図面に記載されたような構成や要素の組合せに限定されて適用されるものではないということは了解されるべきである。本発明は、他の実施形態が可能であり種々の方法で実行され実施され得るものである。また、ここで使われる用語や述語は説明目的のためのものであり発明を限定するものと解釈するべきでないことは了解されるべきである。

10

【 0 0 3 7 】**好みしい実施の形態の詳細な説明**

以下の好みしい実施の形態についての詳細説明中の添付図面について言及する。この図面は本出願の一部をなす。本図面は、説明図という形で、本発明が実行されたときの具体的形態を示すものである。本発明の範囲から外れることなく他の実施形態や構成要素の変更が可能であることは了解されるべきである。

【 0 0 3 8 】

本発明は画像データと共にアプリケーションデータを保存するシステムと方法からなり、概念的に参照番号10に示されている。本方法は、一般に、画像データとアプリケーションデータの収集22、1つの単独JPEGファイルの生成24、及び画像データとアプリケーションデータの表示および／あるいは使用26とで構成される。

20

【 0 0 3 9 】**入力データの収集**

入力データは画像データ100とアプリケーションデータ104とから成る。本発明の1つの用途として、アプリケーションデータ104はオーディオファイルを包含する。他のアプリケーションデータとして、テキスト、ワープロドキュメント、Eメールメッセージ、スプレッドシート、人事ファイル、指紋、その他を含むことができる。アプリケーションデータは当該技術分野で周知のあらゆるデジタルフォーマットであってもよい。

【 0 0 4 0 】

図1の上部に例示したように、画像データとアプリケーションデータを収集するのに用いる装置はデジタルカメラ102である。画像ファイル入力100はJPEG圧縮フォーマットで保存されることが好みしい。カメラ102は、マイクロフォン、アナログからデジタルへのコンバーター、プロセッサー、及びメモリーのような音声をデジタルで記録する装置で構成される。

30

【 0 0 4 1 】

ひとたび画像ファイル入力100と音声ファイル入力104を手に入れると、入力ファイルは単独JPEGファイル32に変換される。

【 0 0 4 2 】**アプリケーションデータのエンコード**

図2に示すように、画像ファイル100とアプリケーションデータ104とを結合して単独JPEGファイル32にするため、アプリケーションデータをエンコードし28、データを解析し29、もし必要ならエンコードされたアプリケーションデータを画像データと結合させエンコードされたアプリケーションデータが組み込まれた単独JPEGファイルを形成する32。アプリケーションデータのエンコーディングステップ28は、一般にバイト0×FFを含むエンコードされていないアプリケーションデータ中のすべてのバイト位置を決定するステップを有し、バイト0×FFのすべての位置を示す少なくとも1つのローケーターバイトを決定し、そのデータからバイト0×FFを除去する。

40

【 0 0 4 3 】

図3に示すように、本発明におけるエンコーディングプロセスは、データのあらかじめ定められたシーケンシャルバイトの数値を評価することから始まる。好みしい実施の形態に

50

おいては、アプリケーションデータファイル 104 から 7 バイトのシーケンシャルデータが評価される。ローケーターバイト 115 が 7 個のシーケンシャルバイト 112 に付け加えられる。デコーディングを簡単にするために、ファイルの記憶において、7 個のシーケンシャルバイトにローケーターバイトを先行させる。ローケーターバイト 115 は、バイト $0 \times FF$ を含むエンコードの前に 7 個のシーケンシャルバイト中のその位置を指示するために使われる。ローケーターバイト 115 と 7 個のシーケンシャルバイトは結合されて 8 バイトのパケットを形成する。

【0044】

図 4 及び図 5 はローケーターバイト 115 の値を決定する 1 つの方法を図示したものである。この方法によれば、マスク値 116 が 7 個のシーケンシャルバイトの各バイト位置に割り付けられる。マスク値 116 の最上位ビット 113 はローケーターバイト 115 が $0 \times FF$ にならないことを確かなものにするため、ゼロにすることが好ましい。演算手順はまず、最初のバイトについてバイト $0 \times FF$ が含まれていないかどうかを決定するためのテストを行う 120。このバイト値が含まれていない場合は、そのテスト結果は「偽」ということになり、ポインターは次のバイトポジションにシフトする 122。この例では、図 3 及び図 4 に示されており、最初のバイトは 0×01 である。従って、テストは「偽」ということになりポインターは 2 番目のバイトに移動し、結果は $0\ 0\ 0\ 0\ 0\ 0\ 0\ 0$ となる。2 番目のバイトは 0×22 なので、ポインターは 3 番目のバイトに移動し、結果は $0\ 0\ 0\ 0\ 0\ 0\ 0$ を維持する。

【0045】

このテストとシフトプロセスはバイトテストが「真」となるか又は 7 個すべてのバイトについてテストが完了するまで繰り返される 128。図 3 及び図 4 に示された例では、ポインターはテスト結果が「真」となるまでに 5 ポジションシフトされるであろう。テストが「真」のとき、演算手順はフラグビット 1 をマスク 116 により示される位置にセットする 126。したがって、結果 117 は $0\ 0\ 0\ 0\ 0\ 1\ 0\ 0$ となる。6 番目のバイト位置では、テストは「偽」となり結果は変化しない。7 番目のバイト位置では、テストは「真」となり結果は $0\ 0\ 0\ 0\ 0\ 1\ 0\ 1$ となる。この結果はローケーターバイト 115 の値として使われ、これは、本例では、16 進数表示での 0×05 に対応する。もし、7 個のシーケンシャルバイトのどれにもバイト $0 \times FF$ が含まれていなければ、ローケーターバイトの値は 0×00 となる。

【0046】

図 3 に示すように、いったんローケーターバイト 115 の値が決定されると、このローケーターバイトはシーケンシャルバイト 112 と結合される。しかし、 $0 \times FF$ を含むどんなバイトも 0×00 に変換され、あらかじめ定められたプレースホールダーバイト 128 となる。 $0 \times FF$ を含まないバイトは同じ値を維持する 130。同時に、これらの 8 バイトはアプリケーションデータのエンコードされたパケットを形成する 131。

【0047】

$0 \times FF$ を除去し、ローケーターバイトを加えたので、 $0 \times FF$ が存在することによるファイルの破壊は避けられ、同時に、元のデータの完全性は保たれる。もちろん、ローケーターバイトの値 115 を決めるために用いることができる別の付番方法や別の演算手順を用いても良い。このような変更も本発明の範囲内である。

【0048】

画像データ 100 に関しては、画像データは、離散コサイン変換 (DCT) とハフマンエンコーディングのような JPEG 方式を使ってエンコードすることができる。

【0049】

エンコードされたパケットの JPEG ファイルへの挿入

エンコードの後、エンコードされたパケット 131 は画像ファイルの位置に挿入され、単独 JPEG ファイルを生成する。図 6 に示すように、JPEG レコードは、最初のバイトが FF (132)、2 番目がヘッダータイプバイト (134)、3 番目 (136) 4 番目のバイト (138) が結合されて、長さバイトを含む 16 ビット長のレコードとなるようなフォー

10

20

30

40

50

マットに従う。エンコードされたパケット131は既存のJPEGパケット140, 144の間に挿入することができる。図7に示すように、JPEGヘッダー132、アプリケーションヘッダータイプ0xE A(134)、10バイトの長さバイト136, 138及び8バイトのエンコードされたアプリケーションデータとから成る新しいJPEGパケットは、2つの既存のJPEGパケット140(図6参照), 144の間に挿入される。

【0050】

当業者は、もしデータがJPEGフォーマットに課せられた65, 535バイトの制限を越える場合、正しい順序にデータを再構成するための仕組みを実行する必要があることを理解するだろう。また当業者は、このようなアプリケーションデータを読み込むシステムは、他のアプリケーションが同じヘッダーコードを使っていた場合でも、既存のデータを信頼度高く検出することができなければならないことを理解するだろう。特にJPEGは16のアプリケーションコードの使用を許すが、JPEG2000は1つだけであることを考慮すれば、その通りである。10

【0051】

本発明は、これらの懸案事項に対して各JPEGアプリケーションレコードの先頭に追加のデータレコードを挿入することで対処している。これらのヘッダーには順序、セキュリティ手段、及び識別データが含まれ信頼度高く元のアプリケーションファイルを再構築できるようにしている。追加されたヘッダー情報はさらに、組み込まれたデータファイルの検出を可能とする役割、元のデータの変更の追跡を可能とする役割、そして再生アプリケーションに使われる情報の提供を行う役割を果たす。追加されたヘッダーはデータ保存構造の体裁を取ることができ、ここではこれをブロックと呼ぶ。ブロックタイプは、JPEGに準拠し、JPEGデータを壊さないことを担保するために、少なくとも最初の12バイトの後は、FFコードを除去することが好ましい。各ブロックにあるバイトの意味及びバイト数はプログラミングの対象に応じて変化させて良いのはもちろんである。ここで表1にインデックスブロックが示される。20

【0052】

【表1】

表1

バイト	説明	値
1-4	JPEGヘッダー、ヘッダータイプ及びサイズ(これはJPEGの具体的な情報)	
5-12	“SOUNDPIX”的ASCII値	SOUNDPIX
13-14	後続のデータブロック数	
15-16	バージョン これはエンコーダーのバージョンである	バイト15は8ビットのメジャーバージョン番号、バイト16は8ビットのマイナーバージョン番号
17	電子透かしフラグ	0=なし；1=デジマーク； 2=Signum；3=SoundPrix
18	暗号化フラグ	0=なし；1=RSA
19	登録フラグ	真(1)なら、エンコーダーツールの登録されたバージョンで作られたことを意味する。
20-21	特徴フラグ(今後の使用のため留保)	
22-24	今後の使用のため留保	
25-N	拡張ファイル情報	

30

40

【0053】

インデックスブロックはファイル全体に適用可能なグローバル情報を含んでいる。バイト50

1 - 1 2 は挿入されたデータを自動的に検出させる。ASCII文字列をレコード内(バイト5 - 1 2)に置くことは、エディターでファイルを見るとき、再生アプリケーションが音声／画像ファイルを特定することを容易にさせる。もし再生アプリケーションが識別レコードを見つけなければ、画像は普通に表示される。もし音声IDが検出されたら、その音声データはヘッダーから読み込まれ、単純な音声ファイルとして再構築される。画像は音声データを取り除くことなく表示される。なぜなら、0 × F F は取り除かれており、現状の復元システムが音声パケットをアプリケーションデータとして特定することができ画像表示に影響を与えることなくスキップすることができるからである。

【0 0 5 4】

画像とデータ文書の結合に何らかの変更がなされていた場合にエラーフラグが表示される
10 ような安全担保及びデジタル透かしをパケット(バイト1 7 及び1 8)に加えても良い。
この機能により暗号キーのようなデータを用いることができ、調査し、探求し、あるいは証拠写真を用いて警察への正式な報告を作成するのに実用的であろう。

【0 0 5 5】

バイト2 0 - 2 1 及び2 2 - 2 4 はオプションであり、将来のプログラム・オブジェクト
20 を完成させるために留保しておく。バイト2 5 - N は、情報の読解能力のあるアプリケーションソフトに、処理を行わせるために拡張された情報を意味する。拡張ファイル情報のフラグタイプは拡張ファイル情報が始まることを明らかにするため0 × F E とするのが好ましい。2番目のバイトは長さフィールドとし0 × F E あるいは0 × F F を含まないよう
20 にするのが好ましい。拡張情報は、情報を読解できないアプリケーションが、従うべき情報のタイプを表示する識別子及びデータを読み飛ばすことができるよう、長さ表示を含むことが好ましい。表2は拡張されたファイル情報のタイプのいろいろな識別子を示す。
すべての識別子は0 × F E に先導されて情報のタグや長さ表示に拡張されるのが好ましい
。。

【0 0 5 6】

【表2】

表2

ID	説明	パラメーター
0 1	オーディオ言語書式	ID の後に、現在の言語書式の合計を与える1バイト が続く。
0 2	テキスト言語書式	ID の後に、現在の言語書式の合計を与える1バイト が続く。
0 3	オーディオ言語書式	1バイトの後に、R F C - 1 7 6 6 言語コードが続 く。このE F I タイプは与えられたファイルに存在す るすべての言語を表示するために用いられる。もし複 数のオーディオ言語が存在する場合は、このファイル は各言語に対し1つの0 3 レコードを持つ。
0 4	テキスト言語書式	1バイトの後に、R F C - 1 7 6 6 言語コードが続 く。このE F I タイプは与えられたファイルに存在す るすべての言語を表示するために用いられる。もし複 数のオーディオ言語が存在する場合は、このファイル は各言語に対し1つの0 4 レコードを持つ。

10

20

30

40

50

【0 0 5 7】

例えば、複数言語のファイルを伴う場合を想定する。0 1 のE F I タイプIDは、ファイル
に複数言語が組み込まれた場合を表示する。0 × 0 1 のタイプは、このファイルに存在す
る相異なる言語の数を記載するバイトが後に続くことを表示する。従って、5言語のファ
イルについての拡張ファイル情報領域は、0 × F E 0 × 0 2 0 × 0 1 0 × 0 5 のよ
うになる。したがって、拡張ファイル情報を用いることは書式に柔軟性を持たせ、将来の
追加事項を扱う上での拡張性を与える。

【 0 0 5 8 】

追加のヘッダーはさらにヘッダーブロックにより成り立つ。ヘッダーブロックはJPEGファイルに含まれるすべてのファイルデータを有している。このデータは、例えば、音声データとスライドショーのための画像データを含む。データには、ワードプロセッシングファイル、スプレッドシート、及び他のランダムなデータもまた含むことができる。ヘッダーブロックとしての基本14バイトは下記表3に示される。

【 0 0 5 9 】**【表3】**

表3

バイト	説明	値
1 - 4	JPEG ヘッダー、ヘッダータイプ及びサイズ（これは JPEG の具体的情報）	
5 - 12	データ書式タグ（上記 SOUNDPIX と類似する、しかし、実際に記憶されているタイプに関連する）。音声情報の場合は、SPXAUDIO、画像情報の場合は、SPXIMAGE、SPXTRADE、及び SPXTHUMB、テキストの場合は、SPXTEXT その他。	SPXIMAGE=画像データ SPXTRADE=商標データ SPXTHUMB=サムネイル画像、 SPXAUDIO = 音声データ、 SPXTEXT = テキストデータ、 SPXEXTRA=ファイルデータ
13 - 14	ID	この ID は音声データヘッダーを一意的に特定し識別するために用いても良い。また、ヘッダーレコードを関連するデータブロックとマッチさせるために用いることができる。ID はデータの再構築を許可する。
15	今後の使用のため留保	
16 - 34	言語	RFC-1766 で規定された言語書式コード
35	音声ファイルフォーマットフラグ	0=wav； 1=au； 2=AIF； 3=MP3；
36 - 38	暗号化フラグ	Hz 単位にて、無署名 24 ビットサンプルレイト
39	音声 bits/sample	無署名 8 bits/sample
40	ディスプレイ優先フラグ ルーピングや自動再生、その他の効果を定義	8 ビットフィールドで 0x01=消音； 0x02=自動再生； 0x04=オンクリック； 0x08=ループ； 0x10=sync (今後の使用のため留保)
41	ループカウント	バイト 40 が 0 8 に設定されると、このバイトが音声を何回ループさせるかを決める。もし 0 に設定されると、音声は連続的にループする。
42 - 43	音声ファイルの開始時間	無署名 16 ビットで 0.1 秒単位の開始時間遅れを表す。また再生間隔における時間遅れにも用いる。
44 - 45	データブロックカウント	このフィールドは、包含する音声ファイルを作るデータブロックの数を示す。
46 - 57	パスワード	音声を聞くために参照する必要のあるエンコードされたパスワード

10

20

30

40

【0060】

データ書式タグ（バイト 5 - 12）はそれに続くデータのタイプを定義する 8 ビットのテキストフィールドを使うことが好ましい。音声データは、SPXAUDIO として表現するのが好ましい。画像データは、SPXIMAGE、SPXTRADE または SPXTHUMB として表現するのが好ましい。テキストデータは、SPXTEXT として表現するのが好ましい。サポートされていないファイルデータは、SPXEXTRA として表現しファイルデータブロックに置くのが好ましい。データ書式タグはさらに多くのデータタイプを組み込むことができ、また、与えられたデータ

50

タイプをどう処理するかを再生システムに知らせる。例えば、MSWORD（登録商標）のデータフォーマットタグのデータと共にワードドキュメントをファイルに組み込むことができる。これは、同一又はそれより高いバージョンの再生装置にMSWORD（登録商標）書式タグを認識させ、データを直接マイクロソフトワードに送信する。

【0061】

さらに画像データヘッダーを含む追加ヘッダーは下記表4に示される。

【表4】

表 4

バイト	説明	
13-14	ID	このタイプの ID は、次のタイプと共に、音声データヘッダーを一意的に識別する役割を果たす。また、ヘッダーレコードを関連するデータブロックとマッチさせるために用いることもできる。ID はデータの再構築を許可する。できれば、1.0 での開始が組み込まれているすべての ID はデータを具備する主 JPEG のため留保しておくのが好ましい。 10
15	画像フォーマット	0=JPEG; 1=BMP; 2=TIFF; 3=GIF
16	画像エフェクト。画像エフェクトは画像を描く又は画像の一部になるような、異なった動作を意味する。さらに、それに関連して画像がサムネイルや商標のデータを有するかどうかを定義する。もし画像エフェクトに商標のサムネイルビットセットがあれば、サムネイルに対してはバイト 23 と 24 に、商標に対しては 25 と 26 に一覧表示された対応する ID が、これらのサムネイル及び商標画像がある場所を示す。画像フォーマットは、さらにマウスを使うなど使用者が対話的に機能させるスライドショーコードがあるかどうかの定義を行う。SPXTRADE と SPXTHUMB に対してはどんなエフェクトも選択できない。	0x00=エフェクトなし 0x01=スライドショー 0x02=マウスオーバー ²⁰ 0x04=サムネイル 0x08=商標
17-18	開始時間。スライドショーと共に使われいつ画像が表示されるかを示す。	無署名 16 ビットで 0.1 秒単位の開始時間遅れを表す。また再生するときの時間遅れにも用いる。SPXTRADE タイプと SPXTHUMB タイプには用いられない。 30
19-20	データブロックカウント	このフィールドは、包含する音声ファイルを作るデータブロックの数を示す。
21-22	関連画像	もしデータフォーマットタグが SPXTRADE または SPXTHUMB に設定された場合、この値は、サムネイルまたは商標が関連する画像の ID を代表する。もしデータフォーマットタグが SPXIMAGE に設定されていた場合、このフィールドは使われない。 40

23-24	サムネイル ID	データフォーマットタグが SPXIMAGE に設定され画像エフェクトにサムネイルセットが含まれている場合にのみ使用される。これは、この画像に関連するサムネイルの ID である。
25-26	商標 ID	データフォーマットタグが SPXIMAGE に設定され画像エフェクトに商標セットが含まれている場合にのみ使用される。これは、この画像に関連する商標の ID である。

10

【0062】

さらにファイルデータヘッダーを含む追加ヘッダーは下記表5に示される。

【表5】

表5

バイト	説明	
13-14	ID	
15	マイムタイプ長、マイムタイプ長はマイムタイプフィールドの長さを定義する。これらの値は標準化され、広く受け入れられており、大きな自由度を持つ。一度この長さを定めると、実際のマイムタイプを展開しデータを読み取ることができる。	
16 から 16+L	マイムタイプ	
17+L から 18+L	データブロックカウント	このフィールドは、包含する音声ファイルを作るデータブロックの数を示す。
19+L	ファイルネーム長	ファイルネームのバイト長
20+L から 20+L+FL	ファイル名	このヘッダーを参照するファイルの名称。

20

30

【0063】

さらにテキストデータヘッダーを含む追加ヘッダーは下記表6に示される。

【表6】

表6

バイト	説明	
13-14	ID	
15-16	関連画像ID。テキストブロック中の関連画像IDはファイル内に含まれる画像をテキストに割り当てる。一例としては、違った時間に多くの画像が表示されるスライドショーの場合である。もしテキストが主画像に割り当てられているのなら、このIDは0がほしい。	10
17-35	言語	RFC-1766で規定された言語書式コード
36-37	データブロックカウント	このフィールドは、包含する音声ファイルを作るデータブロックの数を示す。
38	エフェクト	0x00=エフェクトなし 0x01=マウスオーバー ¹ 0x02=画像アペンド 0x04=オーバーレイ画像 0x08=URL
39-40	オーバーレイ水平オフセット	もしオーバーレイ画像が設定された場合、これはオーバーレイの水平開始位置を設定する。
41-42	オーバーレイ垂直オフセット	もしオーバーレイ画像が設定された場合、これはオーバーレイの垂直開始位置を設定する。

【0064】

さらにユーザーデータブロックを含む追加ヘッダーは下記表7に記載される。ユーザーデータブロックは、ユーザーが各種の情報を蓄えることを可能とする。具体的には、このフィールドタイプは、画像タイトルなどのユーザーが選ぶ情報やメタデータに使うことを想定している。説明フィールドは、開発者がこの領域に蓄えられたデータに意味のある名前を割り当てる事を可能とする。データフィールドには情報を蓄えることがほしい。データフィールドに蓄えられたすべての情報はXMLフォーマットになっていることがほしい。このフォーマットは、この領域に複数のパラメータを蓄えることを可能とし、ユーザーにデータを蓄えるまでの自由度を与える。例えば、ユーザーが「Summer 2001」のタイトルを蓄えようとするとき、データ領域に<Title>Summer 2001</Title>を入力する。この特徴を説明フィールドと共に使えば、ユーザーは関連するデータをグループにして1つのデータブロックにまとめ1つの状態に保持できる。

【0065】

【表7】

30

20

40

表7

バイト	説明	
1-4	JPEG ヘッダー、ヘッダータイプ及びサイズ (これは JPEG の具体的情報)	
5-12	データ書式タグ	SPXUSER
13-14	ID この識別符号は他のロックとの整合性を保つための手段としてのみ用いられる。現在ユーザーデータのデータブロックはない。これは将来の可能性を許すものである。	
15	記述長	
16 から 16+L	説明	保存されたデータのタイプを記述するテキストフィールド
17+L から N	データ	

10

20

30

40

50

【0066】

追加ヘッダーはさらにデータブロックを含む。データブロックはヘッダーブロックに記載されたデータを保存する。下記表8に示すように、IDフィールドを用い、シーケンスフィールドと共に、アプリケーションは関連するヘッダーブロックから与えられた情報をベースに、データを首尾よくデコードするとともに、採るべきアクションを認識する。

【0067】

【表8】

表8

バイト	説明	
1-4	JPEG ヘッダー、ヘッダータイプ及びサイズ (これは JPEG の具体的情報)	
5-12	データ書式タグ	SPXDATA
13-14	ID この ID はヘッダーレコードと整合が取られているのでデータは適当なフォーマットで保存される。	
15 から 16	シーケンス データブロックが収用される元のファイル内のシーケンス	
17 から N	データ	

【0068】

図8に示すとおり、表1-8に記載した追加ヘッダーはJPEGアプリケーションマークー132で始まるJPEGファイルに挿入される。示された実施例では、追加ヘッダーはインデックスヘッダー146とデータ書式148の形をとっている。

【0069】

出力として違った形式のプレゼンティングファイル

組み込まれた単一のJPEGファイル存在すると、それは再生装置26に送信される。再生装置は、パーソナルコンピュータ、携帯情報端末(PDA)、携帯電話、トーキングピクチャーフレームのような特別なアプリケーションプレーヤー、等の少なくとも画像データを表示する能力があるものがほしい。ファイルは、ディスクやEメールあるいは他の情報伝達手段を介して、カメラやディスプレイ装置あるいはアプリケーションコンピュータにも

転送することができる。

【0070】

デコーディングにおいては、再生装置26は組み込まれたデータを検出し、JPEG画像データから切り離し、デコードしてもとのデータフォーマットに戻し、出力として2つのファイルを表示する。図9は本発明のデコード方法を示す。再生装置26はローケーターバイト115をテストすることによりエンコードされたデータの読み込みを開始する(144)。もしローケーターバイト115が0×00なら、再生装置26は続く7バイトを変更せずに転送する146。再生装置26は残りの一連のデータを読み込む。もしローケーターバイト115が0×00でないなら、再生装置26はローケーターバイト115の各ビットをテストする(148)。もしビットが1なら、対応するバイトは0×FFに設定される(150)。もしビットが1でないなら、再生装置26はすべての7ビットがテストされるまで残りのビットをテストする(152, 154)。

10

20

30

40

【0071】

JPEGに組み込まれたアプリケーション

以下に列挙するのはJPEGに組み込まれたランダムなデータに対する実用的ないくつかのアプリケーションである。JPEGに組み込まれたランダムなデータが音声ファイルであるとき、静止画像は、演奏可能な音声ファイルと共に、コンピュータやPDAあるいは携帯電話のような装置に表示することができる。静止画像と音声ファイルとが1つのJPEGに結合されていると、視聴者は、画像として描かれているものについての説明を聞くために、画像をクリックすることができる。音声による説明は、インターネットオークションやカタログサイトからの製品の宣伝や実際の不動産の宣伝、開発活動、家族活動、調査、あるいはニュースレターであっても良い。研究者あるいは調査員は、仲間が成果を見ている間に、所見や分析を音声により聞かせるために、画像ファイルに音声ファイルを付加することができる。画像は、X線のスキャン、超音波スキャン、特別な装置により収集された画像ファイル、その他公知のディジタル画像の形式とすることができます。音声コンテンツを介して画像の検索と類別を行うシステムを提供するため、スピーカーを識別するための音声とスピーカー認識ソフトウェアが本発明に含まれていても良い。本発明のシステムを、ユーザーが、保存されたサムネイルや保存された縮減画像あるいはオリジナル画像から選択できるようにすると共に、ユーザーがオリジナル音声あるいは縮減された音声から選択して演奏するように構成しても良い。

30

【0072】

JPEGでないデータが、違った言語による複合スピーチのような多数の音声ファイルを含んでいる場合、本発明によれば、ユーザーが保存した環境あるいはアクセスを提供したサイトあるいは最も良く使用されている言語のユーザーが使う環境のいずれかにより特定された言語で、ユーザーが音声を保存し演奏することができる。次に、任意のデータが時間に関係付けられたデータを含むとき、音声の再生と同時に特定の時間に多数の画像を保存あるいは表示することができる。あるいは、多数の音声ファイルが特定の開始時刻に再生されるようにしても良い。これはスライドショーを作る上で最も現実的である秘密書類、ID写真、セキュリティ・データ、テキスト、他の画像、個人的なファイル、表計算シート、あるいは指紋によるバイオメトリクスなどの任意のデータは1つの画像データとし、ペーパーレス記録又はペーパーレスデータベースとしても良い。

40

【0073】

任意のデータが、レーザーによるスキャンその他の方法で得た3次元の立体モデルの情報を含んでいる場合、このような任意のデータを、画像として保存されたビジュアルイメージと結合することができ、それにより、立体モデルにビジュアルイメージをマッピングさせ、さらにリアルなモデルにすることができる。

【0074】

結論

本発明は先行技術に関連する多くの問題を解決する。本発明はさらにアプリケーションに関連付けられた新しい独特のJPEGを提供する。本発明は画像ファイルと共に任意のデータ

50

を高信頼度で保存することを可能とする。本発明は画像ファイルとアプリケーションデータとを結合したときに固有のメンテナンスの必要性を低減する。さらに本発明は、コンピュータが入力データに $0 \times FF$ が含まれているアプリケーションデータを読み込んだ場合に修復不可能な障害の発生を防止すると共に、組み込まれたデータの検出を可能とするヘッダーを提供し、オリジナルのアプリケーションファイルの信頼度の高い再構築を可能とし、再生アプリケーションに利用可能な情報を提供し、データに施されたどんな変更も追跡可能としている。

【0075】

以上の通り詳細に説明したが、以上の説明は発明の範囲を制限するものと解釈すべきではなく、単に本発明の現在の最適実施例を説明したものと解釈すべきである。したがって、本発明の範囲は、記載の実施例ではなく添付の請求の範囲及びその法的な均等物により決定されるべきである。10

【図面の簡単な説明】

【0076】

【図1】図1は本発明に係るシステム及び方法を概念的に示す構成図である。

【図2】図2は画像データとアプリケーションデータを含むJPEGファイルの作成過程を概念的に示す構成図である。

【図3】図3は本発明のエンコーディングプロセスの一例を概念的に示した図である

【図4】図4は本発明のエンコーディングプロセスの一例を概念的に示した図である

【図5】図5は本発明の $0 \times FF$ を検出するプロセスを概念的に示したフローチャートである。20

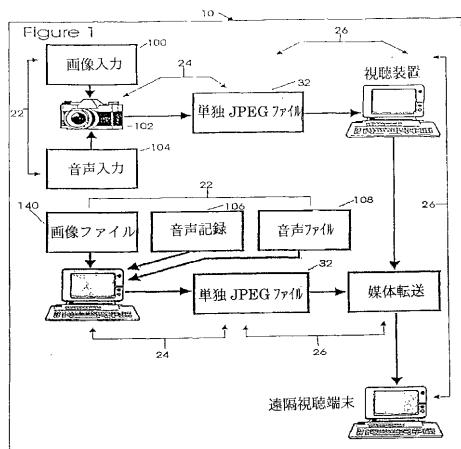
【図6】図6は16進数フォーマットのJPEGファイルの概念的な一例である。

【図7】図7は本発明によりエンコードされたアプリケーションデータが付加されたJPEGファイルの概念的な一例である。

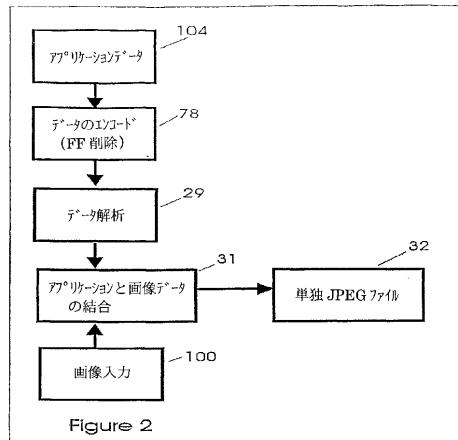
【図8】図8は本発明によりマルチプルエンコードされたアプリケーションセグメントが付加されたJPEGファイルの概念的な一例である。

【図9】図9は本発明によるデコーディングプロセスの概念的な部分である。

【図1】



【図2】

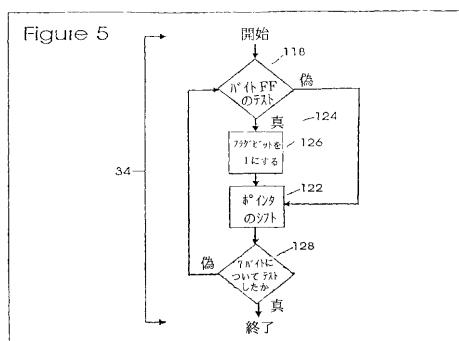


【図4】

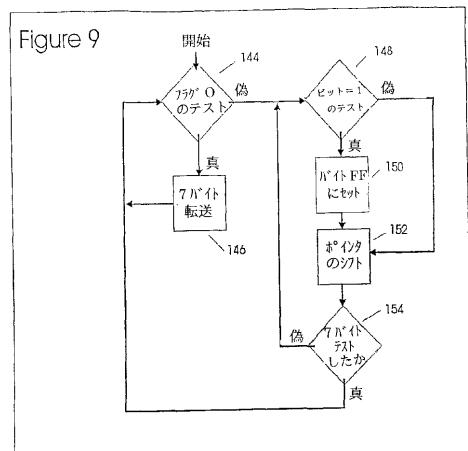
Figure 4

ステップ	マスク	データ	結果
1	0100 0000	0x01	0000 0000
2	0010 0000	0x22	0000 0000
3	0001 0000	0x45	0000 0000
4	0000 1000	0xE0	0000 0000
5	0000 0100	0xFF	0000 0100
6	0000 0010	0xAB	0000 0100
7	0000 0001	0xFF	0000 0101

【図5】



【図9】



【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
26 September 2002 (26.09.2002)

PCT

(10) International Publication Number
WO 02/075640 A2

(51) International Patent Classification: G06K 9/36. (74) Agents: TONG, Rolando, J. et al.; Ian P. Burns & Associates, P.O. Box 20038, Reno, NV 89515-0038 (US)

(21) International Application Number: PCT/US02/08783

(81) Designated States (national): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, IU, 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, SI, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(22) International Filing Date: 19 March 2002 (19.03.2002)

(84) Designated States (regional): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SI, SZ, TZ, UG, ZM, 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, CI, CG, CL, CM, GA, GN, GQ, GW, ML, MR, NU, SN, TD, TG).

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/277,609 19 March 2001 (19.03.2001) US
60/365,203 15 March 2002 (15.03.2002) US

(27) Applicant (for all designated States except US): SOUNDPIX, INC. [US/US]; 689 Bridger Court, Incline Village, NV 89451 (US).

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.



WO 02/075640 A2

(54) Title: SYSTEM AND METHOD OF STORING DATA IN JPEG FILES

(57) **Abstract:** The present invention comprises a system for communicating image data and other data. The system comprises a JPEG file having a JPEG data, at least one non-JPEG data, and a computer. The computer is programmed to read a predetermined number of sequential bytes of the non-JPEG data. The computer is also programmed to determine position of each byte in the sequential bytes that contain a JPEG marker and determine a locator byte, which is capable of indicating the position of any bytes in the sequential bytes that contain a JPEG marker. The computer is further programmed to write the locator byte in the JPEG file and write encoded bytes in the JPEG file. The computer writes the encoded bytes in the JPEG file in a manner wherein, for each byte in the sequential bytes, if the byte does not contain a JPEG marker, each byte is replaced with a bit, preferably written as 0, in the order of the byte's occurrence in the sequential bytes. If the byte does contain a JPEG marker, the byte containing a JPEG marker is replaced with a place holder bit, preferably written as 1 and in the order of the byte's occurrence in the sequential bytes.

SYSTEM AND METHOD OF STORING DATA IN JPEG FILES5 **[01] CROSS REFERENCES TO RELATED APPLICATIONS**

This application claims the benefit of and incorporates by reference, U.S. provisional application number _____, filed on March 15, 2002, and U.S. Provisional application number 60/277,609, filed March 19, 2001.

10 **[02] BACKGROUND OF THE INVENTION****[03] Field of Invention**

[04] The present invention relates to a system and method for storing data in JPEG files.

[05] Description of Related Art**[06] JPEG Files**

15 [07] JPEG is a file format standard used for compressing, storing, and transmitting digital image data. As used herein, the term "JPEG" refers to all versions, revisions, and releases of the format, including "JPEG" and "JPEG 2000." The format was developed by the Joint Photographic Experts Group and it is currently the most popular and widely used image format. One of JPEG's advantages is that it provides the greatest compression of any bitmap format in common use, allowing devices and users to store image files using the least amount of storage space. This compression capability is also one of the features that have made JPEG the most commonly used format for sending pictures over the Internet. Users can transmit and receive images in JPEG more quickly with a higher degree of image quality than in many other formats.

20 [08] The JPEG format uses markers to break files into segments. Markers comprise a first byte having the value 0xFF. JPEG markers can be grouped into two general types; stand-alone and data. A stand-alone marker has no data other than the two bytes of the marker. A

data marker is immediately followed by two bytes that indicate the number of bytes of data that follow the marker. For example, a data marker may appear as follows:

[09] 0xFF 0xNN 0xCH 0xCL

[10] In this example, 0xFF indicates the beginning of a JPEG segment or field, 0xNN denotes the type of field, and 0xCH and 0xCL together indicate the length of the field. The marker may be followed by data or instructions.

[11] In addition to data for the subject image, JPEG allows non-JPEG data or application data to be stored within a JPEG file. The application data may comprise data for applications other than an application that uses image data. For example, data may be inserted into a

10 JPEG file that indicates the setting of the camera that captured the image or the identity of the artist who created the image. In this way, one of the advantages of the JPEG format is that data other than the subject image data can be stored and transmitted in the JPEG file itself.

[12] Embedding random data into a JPEG file offers many practical uses. For example, still images or pictures may be transmitted over the Internet with an accompanying audio file, 15 which may be a person's voice describing the picture. This allows families to share pictures with a family member's voice, realtors to present pictures of a property with a voice description of the property, medical professionals to present diagnostic images with a verbal description or diagnosis, and many other uses.

[13] However, one of the problems associated with storing application data in a JPEG file 20 is the application data cannot contain marker codes. In some instances, decoders can read 0xFF in application data as a JPEG marker and cause irrecoverable failure of the decoding process. If application data in a JPEG file contains a marker code, the marker may cause the JPEG decoder to crash and the file may not be useable.

[14] One solution to this problem is to remove all instances of 0xFF or other marker code 25 from the application data. This would allow the data to be stored and transported in a JPEG

file without the danger of crashing the JPEG decoder. However, if one or more marker codes are removed from the application data, the data is usually significantly altered and it cannot be used by its intended application program.

[15] Therefore, a need exists for a system and method for removing marker codes from application data while the data is stored in a JPEG file and for returning the data to its original state for use with an application program.

[16] Association of Audio and Image Files

[17] Techniques exist for associating random data with JPEG files. For example, users wishing to create digital images with sound can attach a separate audio, e.g., .wav, file to their image file. However, .wav files break down often in that they are easily corrupted or detached during transmission and storage. Another problem with .wav files is that they require special software to be saved and used, such as Microsoft® Media software.

[18] Other techniques for associating random data with JPEG files include techniques disclosed in U.S. patent numbers 5,032,918, issued to Ota et al., and 5,815,201, issued to Hashimoto et al., wherein audio and image files are recorded separately and associated with each other. One of the disadvantages with the invention disclosed in Ota et al. is the necessity of maintaining two separate files. This can be very burdensome and vulnerable to loss of association, especially when high volumes of files are involved or when files are archived for long periods of time.

[19] U.S. patent number 6,102,505 issued to McIntyre et al. creates permanent associations between image and audio data by using invisible ink. U.S. patent number 5,644,557 issued to Akamine et al. and European Patent Application number EP 0 964 304 A1 issued to Nozaki et al. disclose magnetic recording on a film stock and recording data as optical pixels on a film stock, respectively. One of the problems associated with Akamine et al. and Nozaki et al. is the requirement of a physical film stock for storing a hard copy of the audio data, which

is not well suited for digital image data.

Other storage techniques for audio and image data include the invention disclosed in U.S. patent number 5,276,866, issued to Paolini, wherein audio data is stored as image frames in a video stream, which is then compressed. U.S. patent number 5,440,677, issued to Case et al., discloses storing audio files and image files on CD-ROM with additional files to enable associated playback. The problem with Paolini and Case et al., as well as with the above patents, is they require custom programs for both sending and receiving the data before any of the data can be retrieved, displayed, or played.

[20] In sum, a need exists for eliminating file maintenance requirements inherent in associating application data with image data. A need also exists for creating permanent associations with application data and image data in a manner that allows efficient and convenient transmission through a variety of mediums, such as the Internet. A further need exists for efficiently and conveniently combining application data and image data into a single file. A need further exists for storing application data with image data without requiring custom or non-wide used applications for storing, retrieving, displaying, or using application and image data. The present invention provides a system and method that fulfills these needs by approaching the problem in a way that is not suggested by the prior art.

[21] SUMMARY OF INVENTION

20 [22] Advantages of the Invention

[23] An advantage of the present invention is that it allows reliable storage of application data with image data.

[24] Another advantage of the present invention is that it allows application data that may conflict with image decoders to be stored with image data.

25 [25] Another advantage of the present invention is that it eliminates maintenance

requirements inherent in associating application data files with image data files.

[26] Another advantage of the present invention is that it provides a system and method where image data and application data are combined into a single file in a manner that allows application programs to detect the existence of the application.

5 [27] Another advantage of the present invention is that it prevents irrecoverable failure of JPEG decoders when the decoders read application data that contain 0xFF data.

[28] Another advantage of the present invention is that it removes instances of marker codes while at the same time providing data containing order, security, and identification data to enable reliable reconstruction of the original application data.

10 [29] Another advantage of the present invention is that it provides a system and method for storing application data with image data without requiring custom or non-widely used applications for storing, retrieving, displaying, or using application and image data.

[30] Another advantage of the present invention is that it provides a decoding mechanism wherein application data may be reconstructed.

15 [31] Another advantage of the present invention is that it provides an encoding mechanism that inserts additional data records at the beginning of each JPEG application record.

[32] Another advantage of the present invention is that it allows multiple non-application data to be permanently associated with at least one image file.

[33] Another advantage of the present invention is that it encodes original data without 20 losing data.

[34] Another advantage of the present invention is that it preserves original data, including additional information, such as playback parameters, language identification, multiple track information, timing, and the like, during encoding of original data.

25 [35] Another advantage of the present invention is that allows computers to access image data even if the receiving system ignores application data.

[36] Another advantage of the present invention is that it provides digital images with sound.

[37] Another advantage of the present invention is that it provides a means for storing application in an image file in a manner that provides security against modified application
5 data.

[38] Another advantage of the present invention is that it provides a marker in an image file that allows reliable reconstruction of application data.

[39] Another advantage of the present invention is that it provides a means for storing data in an image file that is larger than a maximum field length for the image file.

10 [40] Another advantage of the present invention is that it provides a means for storing data in an image file usable by playback applications.

[41] These and other advantages of the present invention may be realized by reference to the remaining portions of the specification, claims, and abstract.

15 [42] **Brief Description of the Invention**

[43] The present invention comprises a system for communicating image data and other data. The system comprises a JPEG file having a JPEG data, at least one non-JPEG data, and a computer. The computer is programmed to read a predetermined number of sequential bytes of the non-JPEG data. The computer is also programmed to determine position of each
20 byte in the sequential bytes that contain a JPEG marker and determine a locator byte, which is capable of indicating the position of any bytes in the sequential bytes that contain a JPEG marker. The computer is further programmed to write the locator byte in the JPEG file and write encoded bytes in the JPEG file. The computer writes the encoded bytes in the JPEG file in a manner wherein, for each byte in the sequential bytes, if the byte does not contain a JPEG
25 marker, each byte is replaced with a bit, preferably written as 0, in the order of the byte's

occurrence in the sequential bytes. If the byte does contain a JPEG marker, the byte containing a JPEG marker is replaced with a place holder bit, preferably written as 1 and in the order of the byte's occurrence in the sequential bytes.

[44] The above description sets forth, rather broadly, the more important features of the present invention so that the detailed description of the preferred embodiment that follows may be better understood and contributions of the present invention to the art may be better appreciated. There are, of course, additional features of the invention that will be described below and will form the subject matter of claims. In this respect, before explaining at least one preferred embodiment of the invention in detail, it is to be understood that the invention is not limited in its application to the details of the construction and to the arrangement of the components set forth in the following description or as illustrated in the drawings. The invention is capable of other embodiments and of being practiced and carried out in various ways. Also, it is to be understood that the phraseology and terminology employed herein are for the purpose of description and should not be regarded as limiting.

15

[45] **BRIEF DESCRIPTION OF THE DRAWINGS**

[46] Figure 1 is substantially a schematic diagram of the system and method of the present invention.

[47] Figure 2 is substantially a schematic diagram showing creation of a JPEG file containing image and application data.

[48] Figure 3 is substantially an example showing a portion of the encoding process of the present invention.

[49] Figure 4 is substantially an example showing a portion of the encoding process of the present invention.

25 [50] Figure 5 is substantially a flow chart showing 0xFF detection process of the present

invention.

[51] Figure 6 is substantially an example of a JPEG file in hexadecimal format.

[52] Figure 7 is substantially an example of a JPEG file with encoded application data according to the present invention.

5 [53] Figure 8 is substantially an example of a JPEG file with multiple encoded application segments according to the present invention.

[54] Figure 9 is substantially a portion of the decoding process of the present invention.

[55] DESCRIPTION OF THE PREFERRED EMBODIMENT

10 [56] In the following detailed description of the preferred embodiments, reference is made to the accompanying drawings, which form a part of this application. The drawings show, by way of illustration, specific embodiments in which the invention may be practiced. It is to be understood that other embodiments may be utilized and structural changes may be made without departing from the scope of the present invention.

15 [57] The present invention comprises a system and method for storing application data with image data, generally indicated by reference number 10. The method generally comprises gathering image and application data 22, creating a single a single JPEG file 24, and presenting and/or using the image data and application data 26.

[58] **Gathering input data**

20 [59] Input data comprises image data 100 and application data 104. In one use of the present invention, application data 104 comprises audio files. Other application data may include, text, word processing documents, email messages, spreadsheets, personnel files, fingerprints, etc. The application data may be in any digital format known in the art.

[60] In the example shown on the top of figure 1, the device used to gather the image data 25 and application data is a digital camera 102. Image file input 100 is preferably stored in

JPEG compression format. Camera 102 comprises equipment for recording sounds in digital data, such as a microphone, an analog-to-digital converter, processor, and memory.

[61] Once image file 100 and audio file inputs 104 are obtained, input files are converted to a single JPEG file 32.

5 [62] Application Data Encoding

[63] As shown in figure 2, combining image file 100 and application data 104 to a single JPEG file 32, encoding the application data 28, parsing the data 29, if necessary, combining the encoded application data with the image data 31 to form a single JPEG file with embedded encoded application data 32. The step of encoding the application data 28

10 generally comprises determining the location of any bytes in the un-encoded application data that contain a 0xFF byte, determining at least one locator byte that indicates the location of any 0xFF bytes, and removing the 0xFF bytes from the data.

[64] Referring to figure 3, the encoding process of the present invention begins by evaluating a predetermined number of sequential bytes of the data. In the preferred embodiment, seven bytes 112 of sequential data are evaluated from the application data file 104. A locator byte 115 is added to seven sequential bytes 112. To simplify decoding, the locator byte precedes the seven sequential bytes in file storage. Locator byte 115 is used to indicate the position within the seven sequential bytes before encoding that comprise a 0xFF byte. Locator byte 115 and the seven sequential bytes are combined to form an eight byte

15 packet 131.

[65] Figures 4 and 5 illustrate one method by which the value of the locator byte 115 may be determined. In this method, a mask value 116 is assigned to each byte position in the seven sequential bytes. Most significant bit 113 of mask value 116 is preferably zero to ensure that locator byte 115 does not become 0xFF. The algorithm first tests the first byte to

20 determine if it comprises a 0xFF byte 120. If it does not contain this byte value, the test

result is false and the pointer is shifted to the next byte position 122. In the example shown in figures 3 and 4, the first byte is 0xD1. Therefore, the test is false and the pointer is moved to the second byte and the result is 0000 0000. Since the second byte is 0x22, the pointer is shifted to the third byte and the result remains 0000 0000.

5 [66] This test and shift process is repeated until either a byte tests true or all seven bytes have been tested 128. In the example show in figures 3 and 4, the pointer would be shifted to the fifth position before the test result is true. When the test is true, the algorithm sets a flag bit to 1 126 in a position indicated by mask 116. Therefore, the result 117 is 0000 0100. At the sixth byte position, the test is false and the result does not change. At the seventh byte
10 position, the test is true, and the result becomes 0000 0101. This result is used as the value of the locator byte 115, which, in this example, corresponds to 0x05 in hexadecimal format. If none of the seven sequential bytes contained a 0xFF byte, the value of the locator byte would be 0x00.

[67] Referring to figure 3, once the value of locator byte 115 is determined, the locator
15 byte is combined with the sequential bytes 112 of application data. However, any bytes that comprise 0xFF are converted to 0x00, a predetermined place holder byte 128. Any bytes that do not comprise a 0xFF, remain the same 130. Together, these eight bytes form an encoded packet of application data 131.

[68] Having removed 0xFFs and having provided a locator byte, file corruption due to
20 presence of 0xFFs is avoided and, at the same time, integrity of original data is preserved. Of course, different numbering systems may be used or different algorithms may be used to determine the value of locator byte 115. Such modifications are within the scope of the present invention.

[69] With regards to image data 100, the image data may be encoded using the JPEG
25 methods, such as Discrete Cosine Transform (DCT) and Huffman encoding.

[70] **Insertion of Encoded Packet into JPEG file**

[71] After encoding, encoded packet 131 is inserted in a location in the image file to create a single JPEG file. Referring to figure 6, JPEG records follow the format where the first byte is an FF (132), the second is a header type byte (134), the third (136) and fourth bytes (138) are combined to form a 16-bit length of record including length bytes. Encoded packet 131 may be inserted between existing JPEG packets 140, 144. As seen in figure 7, A new JPEG packet 143 comprising of JPEG header 132, application header type 0xEA 134, a length of ten bytes 136, 138, and eight byte encoded application data packet 131 is inserted between two existing JPEG packets 140 (see figure 6), 144.

[72] A person skilled in the art would appreciate that a mechanism must be implemented whereby the data may be reconstructed in the proper order if it exceeds a 65,535-byte segment length limit imposed by the JPEG format. A person skilled in the art would also appreciate that systems reading the application data must be able to reliably detect the existence of the data even if other applications use the same header code. This is true especially considering JPEG allows sixteen application codes to be used, but JPEG2000 allows only one.

[73] The present invention addresses these concerns by inserting additional data records at the beginning of each JPEG application record. These headers contain order, security, and identification data to enable reliable reconstruction of the original application file. Additional header information further serves to enable detection of the embedded data file, provide traceability of any modification to the original data, and provide information usable by playback applications. Additional headers may be in form of data storage structures, hereinafter referred to as blocks. Block types are preferably removed of FF codes at least after the first 12 bytes to ensure data remains compliant with JPEG specification and does not corrupt JPEG data. Designation of bytes and number of bytes reserved for each block may of

course be varied according to programming objective. Referring now to table 1, an index block is shown.

[74] Table 1

5

Byte	Description	Values
1-4	JPG Header, Header Type, and Size (this is JPG Specific information)	
5-12	ASCII value of the word "SOUNDPIX"	SOUNDPIX
13-14	Number of data blocks to follow	
15-16	Version. This is the version of encoder	Byte 15 is an 8-bit major version number. Byte 16 is an 8-bit minor version number.
17	Watermark flag	0 = none; 1 = Digimarc; 2 = Signum; 3 = SoundPix.
18	Encryption Flag	0 = none; 1 = RSA.
19	Registered Flag	If TRUE (1), signifies that this file was created with a registered version of an Encoder Tool
20-21	Profile Information (Reserved for later use)	
22-24	Reserved for later use	
25-N	Extended File Information	

- [75] Index block contains global information that is applicable for the entire file. Bytes 1-12 allow automatic detection of the inserted data. Placing the ASCII string within the record (Bytes 5-12) makes it easy for playback applications to identify an audio/image file when viewing the file in an editor. If the playback application does not find the identification record, the image is displayed normally. If the audio ID is detected, then the audio data is read from the header and reconstructed as a simple audio file. The image is displayed without removal of the audio data. Because the 0xFF's have been removed, all current decompression systems can identify the audio packets as application data and skip them without affecting the image display.

[76] Security and digital watermark may be added to the packets (Bytes 17 and 18) such

that any alteration of the combined image and data document would be flagged as an error.

This feature allows use of data as an encryption key, which may be practical for creating authentic police reports, investigative work, detective work, or using photos for evidentiary purposes.

- 5 [77] Bytes 20-21 and 22-24 are optional and may be reserved to accomplish future programming objectives. Bytes 25-N represent extended information that allow specific applications, which are capable of understanding the information, to process the information. The flag byte for extended file information is preferably a 0xFE to signify beginning of extended information. Second byte is preferably a length field and does not include 0xFE or 10 0xFF. Extended information preferably contains length indicator to allow applications, which cannot understand information, to skip data as well as an identifier to indicate type of information to follow. Table 2 shows various extended file information type identifiers. All identifiers are preferably preceded by the 0xFE extended information tag and a length indicator.

15

[78] Table 2

ID	Description	Parameters
01	Audio Language Formats	1 byte following the ID, giving the count of language formats present.
02	Text Language Formats	1 byte following the ID, giving the count of language formats present.
03	Audio Language Format	1-byte followed by the RFC-1766 Language code. This EFI Type is used to indicate all languages present in a given File. If there are multiple audio languages present, the file should contain a 03 record for each language.
04	Text Language Format	1-byte followed by the RFC-1766 Language code. This EFI Type is used to indicate all languages present in a given File. If there are multiple text languages present, the file should contain a 04 record for each language.

- [79] For example, assume a multiple language file is involved. An EFI Type ID of 01 would indicate multiple languages embedded in the file. A type of 0x01 would indicate following byte describes the number of different language formats present in the file. Thus, Extended File Information area for a file with 5 languages would look like: 0xFE 0x02 0x01 5 0x05. Thus, using extended file information gives the file format the flexibility and extensibility to deal with future additions.
- [80] Additional header further comprises header blocks. Header blocks will contain information about any file data that is to be included in the JPEG file. This data include, as examples, audio data and image data for slide shows. Data may also include word processing files, spreadsheets, and other random data. Base 14 bytes for Header Blocks are shown in 10 table 3 below.

[81] Table 3

Byte	Description	Values
1-4	JPG Header, Header Type, and Size (this is JPG specific information)	
5-12	Data Format Tag (Similar to the SOUNDPIX tag above, but related to the actual type of data stored). For audio information this will be SPXAUDIO, for image information, SPXIMAGE, SPXTRADE, and SPXTHUMB, for text, SPXTEXT, etc.	SPXIMAGE = Image data; SPXTRADE = Trademark Data; SPXTHUMB = Thumbnail Image; SPXAUDIO = Audio data; SPXTEXT = Text data; SPXEXTRA = File data.
13-14	ID.	The ID may be used to uniquely identify and distinguish audio data headers. It may also be used to match header records with their associated data blocks. ID allows for reconstruction of data.
15	Reserved for later use	
16-34	Language	Language format codes as specified in RFC-1766
35	Audio file format flag	0 = wav; 1 = au; 2 = AIFF; 3 = MP3

36-38	Audio sample rate flag	Unsigned 24-bit sample rate in Hz.
39	Audio bits/sample	Unsigned 8-bit bits/sample
40	Display preference flag. This defines such things as looping, auto play or other effects.	8 bit field: 0x01 = mute; 0x02 = auto play; 0x04 = on click; 0x08 = loop; 0x10 = sync (Reserved for later use).
41	Loop Count	If byte 40 is set to 08, this byte will determine how many times the audio will loop. If set to 0, the audio will loop continuously.
42-43	Start time of audio file.	Unsigned 16-bit start delay in deciseconds. Also used for time delay between replays.
44-45	Data Block Count	This field tells the number of data block that make up the included audio file.
46-57	Password	Encoded password that requires verification for the audio to be heard.

The Data Format Tag (bytes 5-12) preferably uses an 8-byte text field to define the type of data to follow. Audio data are preferably represented as SPXAUDIO. Image data are preferably represented as SPXIMAGE, SPXTRADE, or SPXTHUMB. Text data are preferably represented as SPXTEXT. Any unsupported file information are preferably represented as SPXEXTTRA and placed in a File Data block. The data format tag also allows for embedding more data types and allows playback systems to know what to do with a given data type. For example, a word document could be embedded into the file with a Data Format Tag of MSWORD®. This would allow the player of the same version or higher to recognize the MSWORD® format tag, and send the data directly to Microsoft Word.

[82] Additional header further comprises image data header shown in table 4 below.

[83] Table 4

Byte	Description	
13-14	ID	The ID Field for this type, as well as the types to follow, serves to

		uniquely identify and distinguish audio data headers. It may also used to match header records with their associated data blocks. ID allows for reconstruction of data. Preferably, all IDs that are embedded start at 1. 0 will be reserved for the main JPG that contains the data.
15	Image Format	0 = Jpeg; 1 = BMP; 2 = TIFF; 3 = GIF.
16	Image Effect. Image Effects describe different actions that the image can perform or be a part of. It also defines whether or not the image has thumbnail or trademark data associated with it. If the Image effect has the Thumbnail of Trademarks bits set, the corresponding ID listed under bytes 23 and 24 for thumbnails and bytes 25 and 26 for trademarks tells where those thumbnail and trademark images may be located. Image format further defines whether the image is part of a slideshow, of should be activated based on a user interaction, such as mouse-over. No Effect must be selected for all SPXTRADE and SPXTHUMB types.	0x00 = No Effect; 0x01 = Slideshow; 0x02 = Mouse over; 0x04 = Thumbnail; 0x08 = Trademark;
17-18	Start Time. Used with slideshows to tell when image is to be displayed	Unsigned 16-bit start delay in deciseconds. Also used for time delay between replays. Not used for SPXTRADE and SPXTHUMB types
19-20	Data Block Count	This field tells the number of data block that make up the included audio file.
21-22	Associated Image	If the Data Format Tag is set to SPXTRADE or SPXTHUMB, this value represents the ID of Image that the thumbnail or trademark is associated with. If the Data Format Tag is set to SPXIMAGE, this field is not used
23-24	Thumbnail ID	Only used if the Data Format Tag is set to SPXIMAGE and Image Effect has Thumbnail set. This is the ID of a Thumbnail associated with this image.
25-26	Trademark ID	Only used if the Data Format Tag is set to SPXIMAGE and Image Effect has Trademark set. This is

		the ID of a Trademark associated with this image.
--	--	---

[84] Additional header further comprises File Data Header shown and described in table 5 below.

[85] Table 5

5

Byte	Description	
13-14	ID	
15	Mime Type Length (L). Mime Type length defines the length of the Mime Type field. These values are standardized and widely accepted and allow for the most flexibility. Once we have the length, then we can extract the actual mime type and read the file data.	
16 to 16+L	Mime Type	
17+L to 18+L	Data Block Count	This field tells the number of data block that make up the included audio file.
19+L	File Name Length (FL)	Length in bytes of file name
20+L to 20+L+fL	File Name	Name of the file that this header references.

[86] Additional header also comprises text data header shown and described in table 6 below.

[87] Table 6

10

Byte	Description	
13-14	ID	
15-16	Associated Image ID. The Associated Image ID in the Text block allows assignment of text to several images that might be contained within a file. An example of this is in the case of a slide show, where multiple images will be displayed at different times. If the text is to be assigned to only the main image, then its Associated ID is preferably 0.	
17-35	Language	Language format codes as specified in RFC-1766
36-37	Data Block Count	This field tells the number of data

		block that make up the included audio file.
38	Effect	0x00 = No Effect; 0x01 = Mouse Over; 0x02 = Append to Image; 0x04 = Overlay Image; 0x08 = URL
39-40	Horizontal Offset for Overlay	If Overlay Image is set, this sets the horizontal starting point for the overlay.
41-42	Vertical Offset for Overlay	If Overlay Image is set, this sets the vertical starting point for the image.

[88] Additional headers further comprises user data block shown and described in table 7 below. User Data Block allows users to store various types of information. Specific uses for this field type are envisioned to be user preference information and Meta data, such as image titles. The Description field allows developers to assign a meaningful name to the data stored within this area. The Data field preferably stores the information. All information stored within the Data field is preferably XML formatted. This allows for multiple parameters to be stored within this area and gives users added flexibility for the data being stored. For example, a user storing a title that was "Summer 2001" would have an entry in the data area that was: <Title>Summer 2001</Title>. Using this feature along with the Description field would allow a user to group related data together into one data block and keep it together.

[89] Table 7

Byte	Description	
1-4	JPG Header, Header Type, and Size (this is JPG specific information)	
5-12	Data Format Tag	SPXUSER
13-14	ID. This identifier is currently used only as a consistency measure for other blocks. Currently there is no associated data block for user data. This allows for that possibility in future releases	
15	Description Length (L)	

16 to 16+L	Description	Text field describing the type of data stored
17+L to N	Data	

[90] Additional header further comprises data block. Data Block stores data described in the Header Blocks. As shown and described in table 8 below, using the ID field, in conjunction with the Sequence field, an application can successfully decode the data and

- 5 know what action to take, based on the information provided by its associated Header Block.

[91] Table 8

Byte	Description	
1-4	JPG Header, Header Type, and Size (this is JPG specific information)	
5-12	Data Format Tag	SPXDATA
13-14	ID. This ID will match a Header record so the data can be stored in the proper format.	
15-16	Sequence. The sequence in the original file that this data block falls into	
17-N	Data.	

- 10 [92] Referring now to figure 8, additional headers described in tables 1-8 are inserted into a JPEG file beginning with JPBG application marker 132. In the embodiment shown, additional header is in the form of index header 146 and data format tag 148.

[93] **Presenting files of different types as output**

- [94] Once a single embedded JPEG file exists, it may be transferred to any playback system 26, preferably a device capable of at least displaying image data such as a personal computer, a personal digital assistant (PDA), a cell phone, a specialized application player such as a talking picture frame, and the like. The file may also be transferred to any camera, display device, or application computer through any electronic media such as disk, e-mail, or other telecommunications medium.
- 15 [95] During decoding, playback system 26 detects embedded data, separates it from the

JPEG image data, decode it back to original data format, and presents two files as output.

Figure 9 shows decoding method of the present invention. Playback system 26 begins reading encoded data by testing locator byte 115 (144). If locator byte 115 is 0x00, then playback system 26 transfers, without changes, the seven bytes that follow 146. Playback system 26 proceeds to read the rest of data stream. If locator byte 115 is not 0x00, then playback system 26 tests each bit of the locator byte 115 (148). If the bit is 1, then corresponding byte is set to 0xFF (150). If bit is not 1, then playback system 26 proceeds to test the remaining bits until all seven bits are tested (152,154).

[96] Applications of Data Embedded into JPEG

- 10 [97] Listed below are just some of the practical applications for random data embedded into JPEG. When random data embedded into JPEG is an audio file, still pictures may be displayed with playable audio files on devices such as computers, PDAs, or cell phones. With still pictures and audio files combined into a single JPEG file, viewers can click on an image to hear description of what is being portrayed by the image. Audio description may be 15 in a form of an advertisement for products from internet auction or catalogue sites, advertisement for a real estate, investigative work, family activity, research, or newsletters. Researchers or investigators can attach audio files to image files thereby allowing their colleagues to listen to their verbal observations or analyses while looking at their work. Images may be in a form of X-ray scans, ultrasound scans, image files generated from 20 specialized devices, any other types of digital images known in the art. Voice and speaker recognition software to identify the speaker may also be included with the present invention to provide a system that allows searching and categorization of images through their audio content. The system of the present invention may be configured to allow a user select between opening a stored thumbnail or stored reduced image and the original image, while 25 also allowing the user to select between playing the original audio, or a reduced version of the

audio.

[98] When non-JPEG data comprises multiple sound files, such as multiple speeches in different languages, the present invention allows users to store and play sound in a particular language specified by either the stored profile of the user, the site providing access, or profile based on the most popular language users use. Next, when random data comprises time-related data, multiple images may be stored and played at specific times to coincide with the audio playback. Alternatively, multiple audio files may be played with specific start times. This is most practical in creating slideshows.

[99] Random data such as confidential documents, ID photos, security data, text, other images, personnel files, spreadsheets, or fingerprint biometrics may be combined in one image file to build a paperless record or database.

[100] When random data comprises three-dimensional solid modeling information scanned by laser or other methodology, such random data may be combined with the visual image data stored in the image thereby allowing mapping of the visual image onto the solid model, resulting in more realistic models.

[101] CONCLUSION

[102] The present invention solves many of the problems associated with the prior art. The present invention further provides novel and unique JPEG related applications. The present invention allows reliable storage of random data with an image file. It eliminates maintenance requirements inherent in associating application data with image files. It also prevents irrecoverable failure when computer reads application data containing 0xFF data entries while at the same time provides a header that enables detection of embedded data, allows reliable reconstruction of the original application file, provides information usable by playback applications, and provides traceability of any modifications to the data.

[103] Although the description above contains many specifications, these should not be construed as limiting the scope of the invention but as merely providing illustrations of some of presently preferred embodiments of this invention. Thus, the scope of the invention should be determined by the appended claims and their legal equivalents rather than by the examples given.

5

CLAIMS

What is claimed is:

1. A method of storing non-JPEG data in a JPEG file, comprising:
 - (A) reading a predetermined number of sequential bytes of the non-JPEG data;
 - 5 (B) determining a position in the sequential bytes of each byte that contains a JPEG marker;
 - (C) determining a locator byte, wherein the locator byte is capable of indicating the position of each byte in the sequential bytes that contains a JPEG marker; and
 - 10 (D) determining encoded bytes, the encoded bytes comprising:
 - (a) the locator byte;
 - (b) each byte of the sequential bytes that does not contain a JPEG marker; and
 - (c) a place holder byte for each byte of the sequential bytes that comprises a JPEG marker.
2. The method of claim 1, further comprising the step of writing the encoded bytes in a predetermined order.
- 20 3. The method of claim 2, wherein the locator byte is written in the most significant position.
4. The method of claim 2, wherein the sequential bytes are written in order of their occurrence in the sequential bytes and the place holder bytes are written in the order 25 of the bytes they replace.

5. The method of claim 2, wherein the encoded bytes are written in the JPEG file.
 6. The method of claim 5, further comprising writing an application data marker in the
5 JPEG file before the encoded bytes.
 7. The method of claim 1, wherein the predetermined number of sequential bytes is
seven.
- 10 8. The method of claim 1, wherein the JPEG marker comprises 0xFF.
9. The method of claim 1, wherein the place holder byte comprises 0x00.
10. The method of claim 1, wherein the locator byte is between 0x00 to 0x7F.
15
11. The method of claim 1, wherein a locator byte of 0x00 indicates that none of the
sequential bytes contain a JPEG marker.
12. The method of claim 1, wherein the non-JPEG data comprises sound data.
20
13. The method of claim 1, further comprising:
(A) locating an application data marker in the JPEG file;
(B) writing the encoded bytes into the JPEG file after the application data marker.
- 25 14. The method of claim 1, further comprising:

- (A) repeating steps A through D of claim 1 for all of the non-JPEG data;
 - (B) determining a total number of encoded bytes;
 - (C) if the total number of encoded bytes exceeds a maximum JPEG field length:
 - (a) dividing the encoded file into segments, the number of bytes in each segment being less than the maximum JPEG field length; and
 - (b) writing the segments into the JPEG file.
- 5
15. The method of claim 13, further comprising recording the JPEG file with the encoded bytes.
- 10
16. The method of claim 13, further comprising transmitting the JPEG file with the locator byte and the encoded bytes.
17. The method of claim 13, further comprising:
- 15 (A) reading the encoded bytes in the JPEG file;
- (B) determining the locator byte;
- (C) determining the location of any place holder bytes in the encoded bytes from the locator byte; and
- (D) changing each place holder byte to a byte containing JPEG markers, thereby decoding the encoded bytes and obtaining the sequential bytes.
- 20
18. The method of claim 17, further comprising using the sequential bytes in an application program.
- 25 19. The method of claim 17, further comprising recording the sequential bytes.

20. The method of claim 13, further comprising writing control data into the JPEG file,
the control data being relevant to the non-JPEG data.

21. A system for communicating image data and other data, comprising:
- (A) a JPEG file, the JPEG file comprising JPEG data;
 - (B) non-JPEG data;
 - (C) a computer, the computer being programmed to perform the following:
- 5 (a) reading a predetermined number of sequential bytes of the non-JPEG data;
- (b) determining a position in the sequential bytes of each byte that contains a JPEG marker;
- 10 (c) determining a locator byte, wherein the locator byte is capable of indicating the position of each byte in the sequential bytes that contains a JPEG marker;
- (d) determining encoded bytes, the encoded bytes comprising:
 - (i) the locator byte;
 - (ii) each byte of the sequential bytes that does not contain a JPEG marker; and
 - (iii) a place holder byte for each byte of the sequential bytes that comprises a JPEG marker; and
- 15 (e) writing encoded bytes in the JPEG file, the encoded bytes being written in a predetermined order.
- 20 22. The system of claim 21, further comprising a transmission device in communication with the computer, the transmission device being configured to transmit the JPEG file with the encoded bytes.
- 25 23. The system of claim 21, further comprising an image data input device in

communication with the computer, the image data input device being adapted to provide JPEG data.

24. The system of claim 23, wherein the image data input device comprises digital camera.
- 5
25. The system of claim 21, further comprising a non-JPEG data input device in communication with the computer, the non-JPEG data input device being configured to provide non-JPEG data.
- 10
26. The system of claim 21, wherein the non-JPEG data input device comprises a sound recording device.
- 15
27. The system of claim 21, wherein the computer is further programmed to write the encoded bytes in the JPEG file after an application data marker.
28. The system of claim 21, wherein the computer is further programmed to write control data in the JPEG file, the control data being relevant to the non-JPEG data.
- 20
29. The system of claim 21, further comprising a second computer in communication with the computer, the second computer being configured to perform the following:
 - (A) read the encoded bytes in the JPEG file;
 - (B) determine the locator byte;
 - (C) determine the location of any place holder bytes in the encoded bytes from the locator byte; and
- 25

(D) change each place holder byte to a byte containing JPEG markers, thereby decoding the encoded bytes and obtaining the sequential bytes.

30. The method of claim 26, wherein the computer is adapted to use the non-JPEG data.

31. A method of storing non-JPEG data in a JPEG file, comprising:
- (A) reading the non-JPEG data;
 - (B) determining which bytes of the non-JPEG data, if any, comprise a predetermined JPEG marker;
 - 5 (C) determining encoded bytes, the encoded bytes comprising:
 - (a) each of the bytes of the non-JPEG data that do not comprise a JPEG marker;
 - (b) a place holder byte for each byte of the non-JPEG data that comprises the predetermined JPEG marker;
 - 10 (c) at least one locator byte, the locator byte being capable of indicating which of the encoded bytes are place holder bytes.
32. The method of claim 31, further comprising the step of writing the encoded bytes in a predetermined order.
- 15 33. The method of claim 32, wherein the locator byte is written in the most significant position.
34. The method of claim 32, wherein the encoded bytes are written in the JPEG file.
- 20 35. The method of claim 34, further comprising writing an application data marker in the JPEG file before the encoded bytes.
- 25 36. The method of claim 31, wherein the JPEG marker comprises 0xFF.

37. The method of claim 31, wherein the place holder byte comprises 0x00.
38. The method of claim 31, wherein a locator byte of 0x00 indicates that none of the bytes of non-JPEG data contain a JPEG marker.
5
39. The method of claim 31, whercin the non-JPEG data comprises sound data.
40. The method of claim 31, further comprising:
 - (A) locating an application data marker in the JPEG file;
 - 10 (B) writing the encoded bytes into the JPEG file after the application data marker.
41. The method of claim 31, further comprising:
 - (A) determining a total number of encoded bytes;
 - (B) if the total number of encoded bytes exceeds a maximum JPEG field length:
 - 15 (a) dividing the encoded bytes into segments, the number of bytes in each segment being less than the maximum JPEG field length; and
 - (b) writing the segments into the JPEG file.
42. The method of claim 34, further comprising recording the JPEG file with the encoded
20 bytes.
43. The method of claim 34, further comprising transmitting the JPEG file with the locator byte and the encoded bytes.
- 25 44. The method of claim 34, further comprising:

- (A) reading the encoded bytes in the JPEG file;
- (B) determining the locator byte;
- (C) determining the location of any place holder bytes in the encoded bytes from the locator byte; and
- 5 (D) changing each place holder byte to a byte containing JPEG markers, thereby decoding the encoded bytes and obtaining the sequential bytes.

45. The method of claim 34, further comprising using the non-JPEG bytes in an application program.

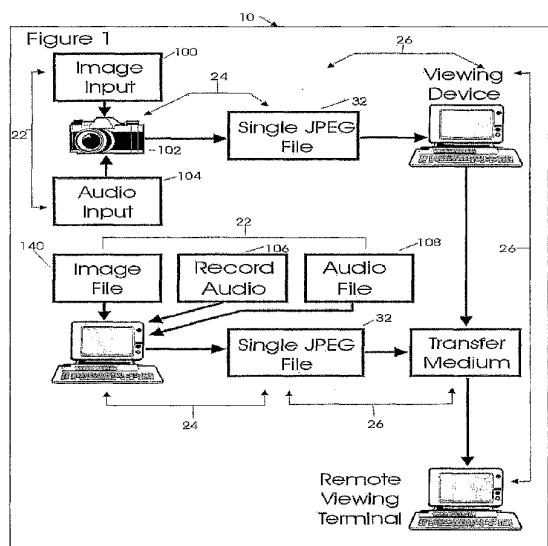
10

46. The method of claim 34, further comprising writing control data into the JPEG file, the control data being relevant to the non-JPEG data.

WO 02/075640

PCT/US02/08783

1/9



SUBSTITUTE SHEET (RULE 26)

WO 02/075640

PCT/US02/08783

2/9

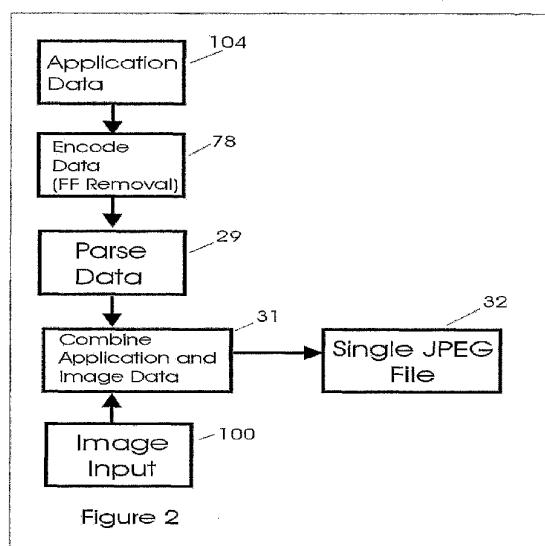


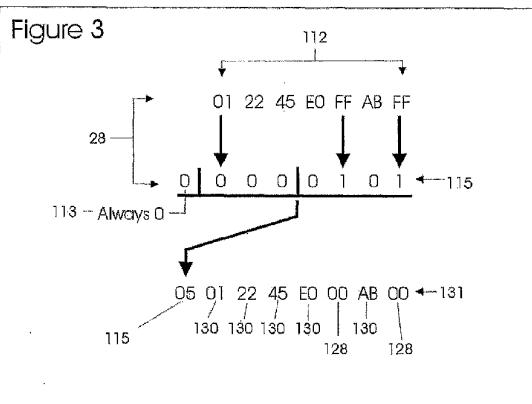
Figure 2

SUBSTITUTE SHEET (RULE 26)

WO 02/075640

PCT/US02/08783

3/9

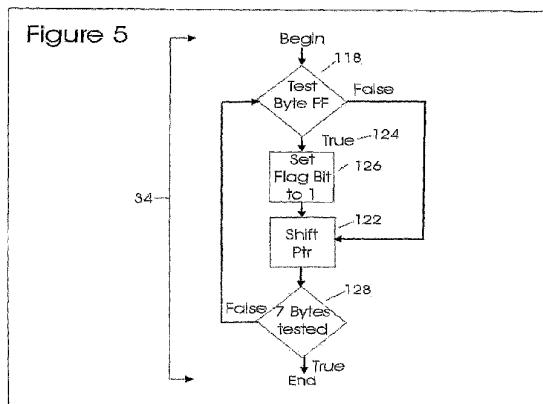


SUBSTITUTE SHEET (RULE 26)

Figure 4

STEP	MASK	DATA	RESULT
1	0100 0000	0x01	0000 0000
2	0010 0000	0x22	0000 0000
3	0001 0000	0x45	0000 0000
4	0000 1000	0xE0	0000 0000
5	0000 0100	0xFF	0000 0100
6	0000 0010	0xAB	0000 0100
7	0000 0001	0xFF	0000 0101

SUBSTITUTE SHEET (RULE 26)



WO 02/075640

PCT/US02/08783

6/9

FIGURE 6

132	134	132		134	136	138
0000	FF DB	FF E0	00 10 4A 46 49 46 C0 01 01 01 01 2C			
0010	01 2C	00 00	FF DB 00 43 00 13 0D 0E 10 0B 0C 13			
0020	10 0F	10 15 14 13 16 1C 2E 1E 1C 1A 1A 1C 39 29				
0030	2B 22 28 43	3B 47 46 42 3B 41 40 4A 54 6B 5A 4A				
0040	4F 65 50 40 41 5D 7E 5E 55 6B 72 77 79 77 48 59					
0050	93 BC 82 74	8B 6B 75 77 73	FF DB 00 43 01 14 15			
	140	144	140			

SUBSTITUTE SHEET (RULE 26)

WO 02/075640

PCT/US02/08783

7/9

FIGURE 7

132 134 136 138 131
0000 **FF DB FF E0** 00 10 4A 46 49 46 0C 01 01 01 01 2C
0010 01 2C 00 00 **FF EA** 00 02 05 01 22 45 **E0** 00 **AB** 00-143
0020 **FF DB** 00 43 00 13 0D 0E 10 0E 0C 13 10 0F 10 15
0030 14 13 16 1C 2E 1E 1C 1A 1A 1C 39 29 2D 22 2E 43
0040 3B 47 46 42 3B 41 40 4A 54 6B 5A 4A 4F 65 50 40
0050 41 5D 7E 5E 65 6E 72 77 79 77 48 59 83 8C 82 74
0060 8B 6B 75 77 73 **FF DB** 00 43 01 14 15
144

SUBSTITUTE SHEET (RULE 26)

WO 02/075640

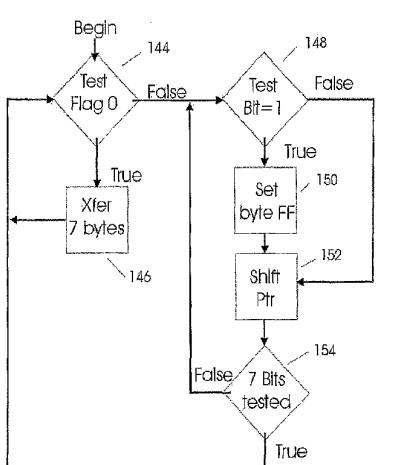
PCT/US02/08783

8/9

FIGURE 8													
148				132			146				132		
0000	FF	D8	FF	E0	00	/10	4A	46	49	46	00	01	01
0010	01	2C	00	00	FF	EA	00	18	53	4F	55	4E	44
0020	00	08	00	00	00	00	00	00	00	00	50	49	58
0030	00	3E	53	50	58	41	55	44	49	4F	00	00	01
0040	00	00	00	00	00	00	00	00	00	00	00	00	00
0050	00	00	00	00	00	00	56	22	08	02	30	00	00
0060	00	00	00	00	00	00	00	00	00	00	00	FF	EA
0070	7C	2F	53	50	58	44	41	54	41	00	00	01	00
0080	49	46	00	46	54	8C	00	00	57	41	00	56	45
											66	6D	74
148													

SUBSTITUTE SHEET (RULE 26)

Figure 9



SUBSTITUTE SHEET (RULE 26)

【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
26 September 2002 (26.09.2002)

PCT

(10) International Publication Number
WO 02/075640 A3(51) International Patent Classification:
G06K 9/36,
G06F 15/16, 17/00, H03M 7/00(72) Inventors; and
(75) Inventors/Applicants (for US only): OSBORNE, John
[US/US]; 793 McCoury Boulevard, Incline Village, NV
89451 (US); RUSSELL, David, W. [US/US]; 575 Sugarpine Drive, Incline Village, NV 89450 (US).

(21) International Application Number: PCT/US02/08783

(22) International Filing Date: 19 March 2002 (19.03.2002)

(25) Filing Language: English

(26) Publication Language: English

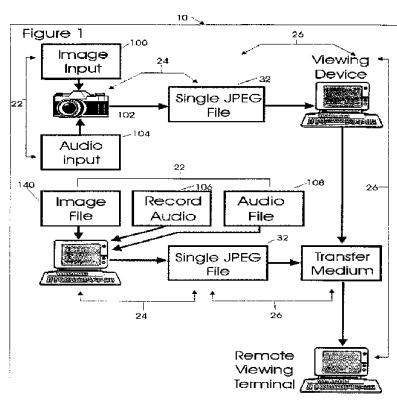
(30) Priority Data:
60/277,609 19 March 2001 (19.03.2001) US
60/365,203 15 March 2002 (15.03.2002) US(71) Applicant (for all designated States except US): SOUND-
PIX, INC. [US/US]; 689 Bridger Court, Incline Village,
NV 89451 (US).(74) Agents: TONG, Rolando, J. et al.; Ian F. Burns & Asso-
ciates, P.O. Box 20038, Reno, NV 89515-0038 (US).(81) Designated States (national): AL, AG, AL, AM, AT, AU,
AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CR, CU, CZ,
DE, DK, DM, DZ, ER, 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, SI, TZ, UG, ZM, ZW),

[Continued on next page]

(54) Title: SYSTEM AND METHOD OF STORING DATA IN JPEG FILES



WO 02/075640 A3



(57) Abstract: The present invention comprises a system for communicating image data and other data. The system comprises a JPEG file (32) having a JPEG data, at least one non-JPEG data (104), and a computer. The computer is programmed to read a predetermined number of sequential bytes of the non-JPEG data (104). The computer is also programmed to determine position of each byte in the sequential bytes that contain a JPEG marker and determine a locator byte, which is capable of indicating the position of any bytes in the sequential bytes that contain a JPEG marker. The computer is further programmed to write the locator byte in the JPEG file (32) and write encoded bytes in the JPEG file (32). The computer writes the encoded bytes in the JPEG file (32) in a manner wherein, for each byte in the sequential bytes, if the byte does not contain a JPEG marker, each byte is replaced with a bit, preferably written as 0, in the order of the byte's occurrence in the sequential bytes. If the byte does contain a JPEG marker, the byte containing a JPEG marker is replaced with a placeholder bit, preferably written as 1 and in the order of the byte's occurrence in the sequential bytes.

WO 02/075640 A3

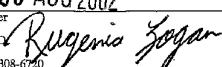
Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM),
European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR,
GB, GR, IT, LU, MC, NL, PT, SI, TR), OAPI patent
(BF, BJ, CF, CG, CL, CM, GA, GN, GQ, GW, ML, MR,
NE, SN, TD, TG).

(88) Date of publication of the international search report:
19 December 2002

Published:
with international search 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.

【国際調査報告】

INTERNATIONAL SEARCH REPORT		International application No. PCT/US02/08783
A. CLASSIFICATION OF SUBJECT MATTER		
IPC(7) : G06F 9/36; G06F 15/16; G06F 17/00; H03M 7/00 US CL : 382/232; 709/217; 707/104.1; 341/107		
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) U.S. : 382/232; 709/217; 707/104.1		
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 practicable, search terms used) EAST		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 5,818,369 A (WITHERS) 06 October 1998, column 5, lines 7-12, 38-46, and 55-56; col. 8, lines 1-4, 35-38, and 45-52; col. 10, lines 27-30; col. 18, lines 47-51. US 5,911,776 A (GUICK) 15 June 1999, col. 6, lines 26-28 and 39-42; col. 13, lines 56-59; col. 14, lines 30-65; Figs. 1 and 8.	1-46
Y		1-46
<input type="checkbox"/> Further documents are listed in the continuation of Box C. <input type="checkbox"/>		See patent family annex.
<p>* Special categories of cited documents:</p> <p>"A" document defining the general state of the art which is not considered to be of particular relevance</p> <p>"B" earlier application or patent published on or after the international filing date</p> <p>"C" document which may cause doubts on priority claimed or which is cited to establish the publication date of another citation or other special reason (as specified)</p> <p>"D" document referring to an oral disclosure, use, exhibition or other means</p> <p>"E" document published prior to the international filing date but later than the priority date claimed</p>		
<p>"F" later documents 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</p> <p>"X" documents 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</p> <p>"Y" documents 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</p> <p>"A," document member of the same patent family</p>		
Date of the actual completion of the international search 15 July 2002 (15.07.2002)	Date of mailing of the international search report 26 AUG 2002	
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PMS-1 Washington, D.C. 20231 Facsimile No. (703)305-3230	Authorized officer ANH HONG DO  Telephone No. 308-6720	

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

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,ZM,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,GQ,GW,ML,MR,NE,SN, TD,TG),AE,AG,AL,AM,AT,AU,AZ,BA,BB,BG,BR,BY,BZ,CA,CH,CN,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

(74)代理人 100118647

弁理士 赤松 利昭

(72)発明者 オズボーン、ジョン

アメリカ合衆国、ネバダ州 89451、インクライン・ビレッジ、マッコリー・ブルバード
793

(72)発明者 ラッセル、ディビッド・ダブリュー

アメリカ合衆国、ネバダ州 89450、インクライン・ビレッジ、シュガーパイン・ドライブ
575

F ターム(参考) 5C053 FA07 GA11 GB21 GB36 KA03 KA24 LA06 LA11
5C078 BA53 BA57 CA14 CA34 DA01 EA01

【要約の続き】

エンコードされたバイトを書きこむ。もしそのバイトがJPEGマークを含んでいる場合は、JPEGマークを含むバイトは、シーケンシャルバイトにおいてこのようなバイトが発生した順に、好ましくは1と書かれたホールダービットに置き換えられる。