

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号

特許第7090730号

(P7090730)

(45)発行日 令和4年6月24日(2022.6.24)

(24)登録日 令和4年6月16日(2022.6.16)

(51)国際特許分類

F I

H 0 4 N 5/92 (2006.01)

H 0 4 N 5/92 0 1 0

H 0 4 N 5/917(2006.01)

H 0 4 N 5/917

H 0 4 N 5/232(2006.01)

H 0 4 N 5/232 3 0 0

H 0 4 N 21/84 (2011.01)

H 0 4 N 21/84

請求項の数 34 (全89頁)

(21)出願番号 特願2020-551333(P2020-551333)

(86)(22)出願日 平成31年4月4日(2019.4.4)

(65)公表番号 特表2021-519525(P2021-519525

A)

(43)公表日 令和3年8月10日(2021.8.10)

(86)国際出願番号 PCT/EP2019/058511

(87)国際公開番号 WO2019/193097

(87)国際公開日 令和1年10月10日(2019.10.10)

審査請求日 令和2年11月24日(2020.11.24)

(31)優先権主張番号 1805711.7

(32)優先日 平成30年4月5日(2018.4.5)

(33)優先権主張国・地域又は機関

英国(GB)

(31)優先権主張番号 1815558.0

(32)優先日 平成30年9月24日(2018.9.24)

最終頁に続く

(73)特許権者 000001007

キヤノン株式会社

東京都大田区下丸子3丁目30番2号

(74)代理人 110003281

特許業務法人大塚国際特許事務所

(72)発明者 ウエドラオゴ, ナエル

フランス国 レンヌ・アタラント, セデ

ックス セッション・セヴィニエ 3 5 5 1

7, リュ ドゥ ラ トゥッシュランペー

ル キヤノン リサーチ センター フラン

ス エス, エー, エス 内

(72)発明者 マゼ, フレデリック

フランス国 レンヌ・アタラント, セデ

ックス セッション・セヴィニエ 3 5 5 1

7, リュ ドゥ ラ トゥッシュランペー

最終頁に続く

(54)【発明の名称】 ファイル内に画像をカプセル化するための方法及び装置

(57)【特許請求の範囲】

【請求項1】

ファイル内に画像をカプセル化する方法であって、前記方法は、
前記画像の撮像モード又は収集モードを識別するグループ化タイプ情報データを生成することと、
前記画像をグループとして記述する構造であって、前記グループ化タイプ情報データ及び前記画像の画像識別子を含むグループ化データ構造を生成することと、
前記グループ化データ構造に関連付けられたプロパティデータ構造を生成することと、
前記グループ化データ構造、前記グループ化タイプ情報データ、前記プロパティデータ構造及び前記画像を前記ファイル内に埋め込むことと、
を含む、方法。

【請求項2】

ファイル内に画像をカプセル化する方法であって、前記方法は、
画像に注釈を付けるための情報を記述するラベルパラメータを決定することと、
前記画像をグループとして記述する構造であって、前記ラベルパラメータ及び前記画像の画像識別子を含むグループ化データ構造を生成することと、
前記グループ化データ構造に関連付けられたプロパティデータ構造を生成することと、
前記グループ化データ構造、前記ラベルパラメータ、前記プロパティデータ構造及び前記画像を前記ファイル内に埋め込むことと、
を含む、方法。

【請求項 3】

ファイル内の画像を読み取る方法であって、前記方法は、
前記画像の撮像モード又は収集モードを識別するグループ化タイプ情報データを読み取る
ことと、
前記画像をグループとして記述する構造であって、前記グループ化タイプ情報データ及び
前記画像の画像識別子を含むグループ化データ構造を読み取ることと、
前記グループ化データ構造に関連付けられたプロパティデータ構造を読み取ることと、
前記グループ化データ構造内で識別される前記画像を読み取ることと、
を含む、方法。

【請求項 4】

ファイル内の画像を読み取る方法であって、前記方法は、
画像に注釈を付けるための情報を記述するラベルパラメータを読み取ることと、
前記画像をグループとして記述する構造であって、前記ラベルパラメータ及び前記画像の
画像識別子を含むグループ化データ構造を読み取ることと、
前記グループ化データ構造に関連付けられたプロパティデータ構造を読み取ることと、
前記グループ化データ構造内で識別される前記画像を読み取ることと、
を含む、方法。

【請求項 5】

データ構造は、ISOベースメディアファイルフォーマットで定義されるボックスである
、請求項 1 乃至 4 の何れか 1 項に記載の方法。

【請求項 6】

グループ化タイプ情報データが、オート露光ブラケットモード、ホワイトバランスブラケ
ットモード、フォーカスブラケットモード、フラッシュ露光ブラケットモード、被写界深
度ブラケットモード、ISOブラケットモード、タイムラプスモード、パノラマモード、
画像バーストモード、ユーザコレクションモード、超解像モード、多重露光モード、ノイ
ズ低減モード、長秒時露光ノイズ低減モード、口径食補償モード、及び/又はHDR撮像
モードを含む前記撮像モードのうちの少なくとも1つに対して定義される、請求項 1 又は
3 に記載の方法。

【請求項 7】

前記グループ化データ構造は、ISOベースメディアファイルフォーマットで定義される
EntityToGroupボックス、SampleToGroupボックス、又はTrackGroupボックスのうちの1つである、請求項 1 に記載の方法。

【請求項 8】

前記プロパティデータ構造は、ISOベースメディアファイルフォーマットで定義される
SampleEntryボックス、SampleGroupEntryボックス、ItemFullPropertyボックス、又はItemPropertyボックスのうちの
1つである、請求項 1 又は 3 に記載の方法。

【請求項 9】

ファイル内に画像をカプセル化する装置であって、前記装置は、
前記画像の撮像モード又は収集モードを識別するグループ化タイプ情報データを生成する
ことと
前記画像をグループとして記述する構造であって、前記グループ化タイプ情報データ及び
前記画像の画像識別子を含むグループ化データ構造を生成することと、
前記グループ化データ構造に関連付けられたプロパティデータ構造を生成することと、
前記グループ化データ構造、前記グループ化タイプ情報データ、前記プロパティデータ構
造及び前記画像をファイル内に埋め込むことと、
のために構成される回路を含む、装置。

【請求項 10】

ファイル内に画像をカプセル化する装置であって、前記装置は、
画像に注釈を付けるための情報を記述するラベルパラメータを決定することと、

10

20

30

40

50

前記画像をグループとして記述する構造であって、前記ラベルパラメータ及び前記画像の画像識別子を含むグループ化データ構造を生成することと、

前記グループ化データ構造に関連付けられたプロパティデータ構造を生成することと、
前記グループ化データ構造、前記ラベルパラメータ、前記プロパティデータ構造及び前記画像を前記ファイル内に埋め込むことであって、前記ラベルパラメータは前記ファイルのメタデータ部に埋め込まれる、埋め込むことと、
のために構成される回路を含む、装置。

【請求項 1 1】

ファイル内の画像を読み取る装置であって、前記装置は、
前記画像の撮像モード又は収集モードを識別するグループ化タイプ情報データを読み取る
ことと、

10

前記画像をグループとして記述する構造であって、前記グループ化タイプ情報データ及び
前記画像の画像識別子を含むグループ化データ構造を読み取ることと、

前記グループ化データ構造に関連付けられたプロパティデータ構造を読み取ることと、
前記グループ化データ構造内で識別される前記画像を読み取ることと、
のために構成される回路を含む、装置。

【請求項 1 2】

ファイル内の画像を読み取る装置であって、前記装置は、
画像に注釈を付けるための情報を記述するラベルパラメータを読み取ることと、
前記画像をグループとして記述する構造であって、前記ラベルパラメータ及び前記画像の
画像識別子を含むグループ化データ構造を読み取ることと、

20

前記グループ化データ構造に関連付けられたプロパティデータ構造を読み取ることと、
前記グループ化データ構造内で識別される前記画像を読み取ることと、
のために構成される回路を含む、装置。

【請求項 1 3】

ファイル内にメディアデータをカプセル化する方法であって、前記方法は、
前記メディアデータの少なくとも一部に対応するエンティティのグループを記述するグルー
ープ化データ構造を生成することと、

30

エンティティのグループに関連付けられたプロパティを包含するプロパティコンテナデー
タ構造を生成することと、

前記プロパティと前記グループ化データ構造との間の関連付け情報を含む関連付けデータ
構造を生成することと、

前記プロパティコンテナデータ構造と前記関連付けデータ構造とを含むプロパティデータ
構造を生成することと、

前記グループ化データ構造、前記プロパティデータ構造、及び前記メディアデータを前記
ファイル内に埋め込むことと、

を含む、方法。

【請求項 1 4】

エンティティのグループに関連付けられた少なくとも 1 つのテキスト属性と、前記少なく
とも 1 つのテキスト属性に関連付けられた言語属性と、を含むテキストプロパティを、前
記プロパティコンテナデータ構造内に生成する、請求項 1 3 に記載の方法。

40

【請求項 1 5】

前記テキストプロパティは、複数の言語属性のそれぞれに関連付けられた複数の同じテキ
スト属性を含む、請求項 1 4 に記載の方法。

【請求項 1 6】

前記プロパティコンテナデータ構造は第 1 及び第 2 のテキストプロパティを含み、前記第
1 のテキストプロパティは第 1 の言語属性に関連付けられたテキスト属性を含み、前記第
2 のテキストプロパティは第 2 の言語属性に関連付けられたテキスト属性を含み、前記第
1 の言語属性は前記第 2 の言語属性とは異なり、前記関連付けデータ構造は、前記第 1 及
び第 2 のテキストプロパティのそれぞれと前記エンティティのグループとの間の関連付け

50

情報と、前記グループ化データ構造とを含む、請求項 1 4 に記載の方法。

【請求項 1 7】

前記テキストプロパティは、名前テキスト属性、記述テキスト属性、及びタグテキスト属性を含む複数のテキスト属性を含む、請求項 1 4 に記載の方法。

【請求項 1 8】

前記プロパティコンテナデータ構造内で、名前テキストプロパティ、記述テキストプロパティ、及び / 又はタグテキストプロパティのうちの少なくとも 1 つを生成することであって、前記テキストプロパティのそれぞれはエンティティのグループに関連付けられたテキスト属性と前記テキスト属性に関連付けられた言語属性とを少なくとも含む、生成すること、を含む、請求項 1 4 に記載の方法。

10

【請求項 1 9】

前記テキストプロパティそれぞれは、複数の言語属性のそれぞれに関連付けられた複数の同じテキスト属性を含む、請求項 1 5 に記載の方法。

【請求項 2 0】

前記関連付けデータ構造は、少なくともエンティティとのプロパティの関連付けをさらに含む、請求項 1 3 乃至 1 9 の何れか 1 項に記載の方法。

【請求項 2 1】

ファイル内のメディアデータを読み取る方法であって、前記方法は、
前記メディアデータの少なくとも一部に対応するエンティティのグループを記述するグループ化データ構造を読み取ることと、
エンティティのグループに関連付けられたプロパティを包含するプロパティコンテナデータ構造を読み取ることと、
前記プロパティと前記グループ化データ構造との間の関連付け情報を含む関連付けデータ構造を読み取ることと、
前記プロパティコンテナデータ構造と前記関連付けデータ構造とを含むプロパティデータ構造を読み取ることと、
前記グループ化データ構造内で識別される前記メディアデータを読み取ることと、
を含む、方法。

20

【請求項 2 2】

エンティティのグループに関連付けられた少なくとも 1 つのテキスト属性と、前記少なくとも 1 つのテキスト属性に関連付けられた言語属性と、を含むテキストプロパティを、前記プロパティコンテナデータ構造内から読み取る、請求項 2 1 に記載の方法。

30

【請求項 2 3】

前記テキストプロパティは、複数の言語属性のそれぞれに関連付けられた複数の同じテキスト属性を含む、請求項 2 2 に記載の方法。

【請求項 2 4】

前記テキストプロパティは、名前テキスト属性、記述テキスト属性、及びタグテキスト属性を含む複数のテキスト属性を含む、請求項 2 3 に記載の方法。

【請求項 2 5】

前記プロパティコンテナデータ構造内で、名前テキストプロパティ、記述テキストプロパティ、及び / 又はタグテキストプロパティのうちの少なくとも 1 つを読み取ることであって、前記テキストプロパティのそれぞれはエンティティのグループに関連付けられたテキスト属性と前記テキスト属性に関連付けられた言語属性とを少なくとも含む、読み取ること、を含む、請求項 2 2 に記載の方法。

40

【請求項 2 6】

前記テキストプロパティそれぞれは、複数の言語属性のそれぞれに関連付けられた複数の同じテキスト属性を含む、請求項 2 5 に記載の方法。

【請求項 2 7】

前記プロパティコンテナデータ構造は第 1 及び第 2 のテキストプロパティを含み、前記第 1 のテキストプロパティは第 1 の言語属性に関連付けられたテキスト属性を含み、前記第

50

2 のテキストプロパティは第 2 の言語属性に関連付けられたテキスト属性を含み、前記第 1 の言語属性は前記第 2 の言語属性とは異なり、前記関連付けデータ構造は、前記第 1 及び第 2 のテキストプロパティのそれぞれと前記エンティティのグループとの間の関連付け情報と、前記グループ化データ構造とを含む、請求項 2 5 に記載の方法。

【請求項 2 8】

前記関連付けデータ構造は、少なくともエンティティとのプロパティの関連付けをさらに含む、請求項 2 1 乃至 2 7 の何れか 1 項に記載の方法。

【請求項 2 9】

ファイル内にメディアデータをカプセル化する装置であって、前記装置は、
前記メディアデータの少なくとも一部に対応するエンティティのグループを記述するグループ化データ構造を生成することと、

10

エンティティのグループに関連付けられたプロパティを含むプロパティコンテナデータ構造を生成することと、

前記プロパティと前記グループ化データ構造との間の関連付け情報を含む関連付けデータ構造を生成することと、

前記プロパティコンテナデータ構造と前記関連付けデータ構造とを含むプロパティデータ構造を生成することと、

前記グループ化データ構造、前記プロパティデータ構造、及び前記メディアデータを前記ファイル内に埋め込むことと、

のために構成される回路を含む、装置。

20

【請求項 3 0】

ファイル内にメディアデータをカプセル化する装置であって、前記装置は、

前記メディアデータの少なくとも一部に対応するエンティティのグループを記述するグループ化データ構造を生成することと、

エンティティのグループに関連付けられたテキスト属性と前記テキスト属性に関連付けられた言語属性とを少なくとも含む前記グループ化データ構造内にテキストプロパティを生成することと、

前記メディアデータと前記グループ化データ構造とを前記ファイル内に埋め込むことと、

のために構成される回路を含む、装置。

【請求項 3 1】

30

ファイル内のメディアデータを読み取る装置であって、前記装置は、

前記メディアデータの少なくとも一部に対応するエンティティのグループを記述するグループ化データ構造を読み取ることと、

エンティティのグループに関連付けられたプロパティを含むプロパティコンテナデータ構造を読み取ることと、

前記プロパティと前記グループ化データ構造との間の関連付け情報を含む関連付けデータ構造を読み取ることと、

前記プロパティコンテナデータ構造と前記関連付けデータ構造とを含むプロパティデータ構造を読み取ることと、

前記グループ化データ構造内で識別される前記メディアデータを読み取ることと、

40

のために構成される回路を含む、装置。

【請求項 3 2】

ファイル内のメディアデータを読み取る装置であって、前記装置は、

前記メディアデータの少なくとも一部に対応するエンティティのグループを記述するグループ化データ構造を読み取ることと、

エンティティのグループに関連付けられたテキスト属性と前記テキスト属性に関連付けられた言語属性とを少なくとも含む前記グループ化データ構造内のテキストプロパティを読み取ることと、

前記グループ化データ構造内で識別される前記メディアデータを読み取ることと、

のために構成される回路を含む、装置。

50

【請求項 3 3】

請求項 1 乃至 8 及び 1 3 乃至 2 8 の何れか 1 項に記載の方法を実装するためのコンピュータプログラムの命令を格納する、コンピュータ可読記憶媒体。

【請求項 3 4】

実行されると、請求項 1 乃至 8 及び 1 3 乃至 2 8 の何れか一項に記載の方法を行わせる、コンピュータプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本開示は、ファイル内に複数の画像をカプセル化するための方法及び装置に関する。

10

【背景技術】

【0002】

最新のカメラでは、画像を撮像するためにさまざまな撮像モードが提供されている。これらの撮像モードのいくつかによれば、一連の画像が撮像される。例えば、それらのカメラはいくつかの画像が撮像されるブラケットモードを提供し、撮像の 1 つのパラメータの値は撮像画像毎に変化する。このパラメータは例えば、露光時間、ホワイトの値、又はフォーカスであってもよい。画像バーストモードは、遅延なしに一連の画像を撮る能力を提供する。これは、例えば、スポーツにおける高速イベントを撮像するために使用することができる。パノラマモードは、シーンの大きなビューを再構成するために一連の重なり合う画像を取得することを可能にする。

20

【0003】

カメラで撮像された画像は、例えばメモ리카ード等の記憶装置に格納される。この画像は、典型的には記憶装置上のデータのサイズを縮小するために符号化される。J P E G やより最近の H E V C 規格のように、多くの符号化規格が使われてもよい。

【0004】

H E V C 規格は静止画像の符号化のためのプロファイルを定義し、単一の静止画像又は静止画像のバーストを圧縮するための特定のツールを記述する。このような種類の画像データに使用される I S O ベースメディアファイルフォーマット (I S O B M F F) の拡張が、「H E I F 又は High Efficiency Image File Format」という名称で I S O / I E C 2 3 0 0 8 規格の第 1 2 部に含めるために提案されている。

30

【0005】

H E I F (High Efficiency Image File Format) は、画像及び画像シーケンスの記憶と共有とのために Moving Picture Experts Group (M P E G) によって開発された規格である。

【0006】

M I A F (Multi-Image Application Format) は M P E G によって I S O / I E C 2 3 0 0 0 規格部 2 2 に開発された規格であり、高効率画像ファイル (H E I F) フォーマットに埋め込まれた画像の作成、読み取り、解析、及び復号のための正確な相互運用性点に対する H E I F 仕様の制約のセットを定義する。

【発明の概要】

40

【0007】

符号化画像のプロパティを記述する限定されたグループ化機構及び限定された機構を提供する一方で、H E I F 及び M I A F ファイルフォーマットは、引用された撮像モード若しくは収集モードの一つに従う撮像又は編集の結果として生じる画像及び画像のグループを収集及び記述するように適合された効率的なグループ化及びプロパティ記述機構を提供しない。

【0008】

本発明は、前述の問題のうちの 1 つ以上を対処するように考案された。本発明は、複数の画像をもたらす撮影モード及び編集モードに適合した、H E I F におけるグループ化とプロパティメカニズムとの拡張に関する。

50

【 0 0 0 9 】

本発明の別の態様によれば、ファイル内に画像をカプセル化する方法であって、
画像の撮像モードに基づくグループ化タイプを決定することと、
画像をグループとして記述するグループ化データ構造を生成することであって、グループ
化データ構造はグループの画像の画像識別子を含む、生成することと、
グループ化データ構造、グループ化タイプ、及び画像をファイル内に埋め込むことと、を
含む方法が提供される。

【 0 0 1 0 】

一実施形態では、グループ化タイプはグループ化データ構造内に含まれる。

【 0 0 1 1 】

一実施形態では、
グループ化データ構造は汎用撮像モードタイプを含み、
グループ化タイプは、汎用撮像モードタイプを特徴付けるために、グループの画像に関連
付けられたプロパティデータ構造内に含まれる。

【 0 0 1 2 】

一実施形態では、プロパティデータ構造はグループ化タイプに関連付けられたパラメータ
をさらに含む。

【 0 0 1 3 】

一実施形態では、
グループ化データ構造は、汎用撮像モードタイプを含み、
グループ化タイプは、追加のパラメータとしてグループ化データ構造内に含まれる。

【 0 0 1 4 】

一実施形態では、グループ化データ構造に関連付けられたプロパティデータ構造が、ファ
イル内にさらに埋め込まれる。

【 0 0 1 5 】

本発明の別の態様では、ファイル内に画像をカプセル化する方法であって、
画像に注釈を付けるための情報を記述するラベルパラメータを決定することと、
画像をグループとして記述するグループ化データ構造を生成することであって、グループ
化データ構造はグループの画像の画像識別子を含む、生成することと、
グループ化データ構造、ラベルパラメータ、及び画像をファイル内に埋め込むことと、を
含む方法が提供される。

【 0 0 1 6 】

本発明の別の態様では、ファイル内に画像をカプセル化する方法であって、
画像の撮像モードに基づくグループ化タイプを決定することと、
第1の画像に関連付けられた参照データ構造を生成し、第2の画像を参照することであっ
て、参照データ構造はグループ化タイプを含む、生成することと、
参照データ構造及び画像をファイル内に埋め込むことと、を含む方法が提供される。

【 0 0 1 7 】

本発明の別の態様では、ファイル内の画像を読み取る方法であって、
画像の撮像モードに基づくグループ化タイプを読み取ることと、
画像をグループとして記述するグループ化データ構造を読み取ることであって、グループ
化データ構造はグループの画像の画像識別子を含む、読み取ることと、
グループ化タイプに従うグループ化データ構造内で識別される画像を読み取ることと、を
含む方法が提供される。

【 0 0 1 8 】

一実施形態では、グループ化タイプはグループ化データ構造内に含まれる。

【 0 0 1 9 】

一実施形態では、
グループ化データ構造は汎用撮像モードタイプを含み、
グループ化タイプは、汎用撮像モードタイプを特徴付けるために、グループの画像に関連

10

20

30

40

50

付けられたプロパティデータ構造内に含まれる。

【0020】

一実施形態では、プロパティデータ構造はグループ化タイプに関連付けられたパラメータをさらに含む。

【0021】

一実施形態では、グループ化データ構造は、汎用撮像モードタイプを含み、グループ化タイプは、追加のパラメータとしてグループ化データ構造内に含まれる。

【0022】

一実施形態では、グループ化データ構造に関連付けられたプロパティデータ構造がファイル内にさらに埋め込まれる。

【0023】

本発明の別の態様では、ファイル内の画像を読み取る方法であって、画像に注釈を付けるための情報を記述するラベルパラメータを読み取ることと、画像をグループとして記述するグループ化データ構造を読み取ることであって、グループ化データ構造はグループの画像の画像識別子を含む、読み取ることと、ラベルパラメータに従ってグループ化データ構造内で識別される画像を読み取ることと、を含む方法が提供される。

【0024】

本発明の別の態様では、ファイル内の画像を読み取る方法であって、画像の撮像モードに基づくグループ化タイプを読み取ることと、第1の画像に関連付けられた参照データ構造を読み取り、第2の画像を参照することであって、参照データ構造はグループ化タイプを含む、読み取ることと、ファイル内の第1及び第2の画像を読み取ることと、を含む方法が提供される。

【0025】

一実施形態では、データ構造は、ISOベースメディアファイルフォーマットで定義されるボックスである。

【0026】

一実施形態では、グループ化タイプが、オート露光ブラケットモード、ホワイトバランスブラケットモード、フォーカスブラケットモード、フラッシュ露光ブラケットモード、被写界深度ブラケットモード、ISOブラケットモード、タイムラプスモード、パノラマモード、画像バーストモード、ユーザコレクションモード、超解像モード、多重露光モード、ノイズ低減モード、長秒時露光ノイズ低減モード、口径食補償モード、及び/又はHDR撮像モードを含む撮像モードのうちの少なくとも1つに対して定義される。

【0027】

一実施形態では、グループ化データ構造は、ISOベースメディアファイルフォーマットで定義されるEntityToGroupボックス、SampleToGroupボックス、又はTrackGroupボックスのうちの1つである。

【0028】

一実施形態では、プロパティデータ構造は、ISOベースメディアファイルフォーマットで定義されるSampleEntryボックス、SampleGroupEntryボックス、ItemFullPropertyボックス、又はItemPropertyボックスのうちの1つである。

【0029】

本発明の別の態様では、ファイル内に画像をカプセル化する装置であって、画像の撮像モードに基づくグループ化タイプを決定することと、画像をグループとして記述するグループ化データ構造を生成することであって、グループ化データ構造はグループの画像の画像識別子を含む、生成することと、グループ化データ構造、グループ化タイプ、及び画像をファイル内に埋め込むことと、のために構成される回路を含む装置が提供される。

【0030】

10

20

30

40

50

本発明の別の態様では、ファイル内に画像をカプセル化する装置であって、画像に注釈を付けるための情報を記述するラベルパラメータを決定することと、画像をグループとして記述するグループ化データ構造を生成することであって、グループ化データ構造はグループの画像の画像識別子を含む、生成することと、グループ化データ構造、ラベルパラメータ、及び画像をファイル内に埋め込むことと、のために構成される回路を含む装置が提供される。

【 0 0 3 1 】

本発明の別の態様では、ファイル内に画像をカプセル化する装置であって、画像の撮像モードに基づくグループ化タイプを決定することと、第 1 の画像に関連付けられた参照データ構造を生成し、第 2 の画像を参照することであって、参照データ構造はグループ化タイプを含む、生成することと、参照データ構造及び画像をファイル内に埋め込むことと、のために構成される回路を含む装置が提供される。

10

【 0 0 3 2 】

本発明の別の態様では、ファイル内の画像を読み取る装置であって、画像の撮像モードに基づくグループ化タイプを読み取ることと、画像をグループとして記述するグループ化データ構造を読み取ることであって、グループ化データ構造はグループの画像の画像識別子を含む、読み取ることと、グループ化タイプに従ってグループ化データ構造内で識別される画像を読み取ることと、のために構成される回路を含む装置が提供される。

20

【 0 0 3 3 】

本発明の別の態様では、ファイル内の画像を読み取る装置であって、画像に注釈を付けるための情報を記述するラベルパラメータを読み取ることと、画像をグループとして記述するグループ化データ構造を読み取ることであって、グループ化データ構造はグループの画像の画像識別子を含む、読み取ることと、ラベルパラメータに従ってグループ化データ構造内で識別される画像を読み取ることと、のために構成される回路を含む装置が提供される。

【 0 0 3 4 】

本発明の別の態様では、ファイル内の画像を読み取る装置であって、画像の撮像モードに基づくグループ化タイプを読み取ることと、第 1 の画像に関連付けられた参照データ構造を読み取り、第 2 の画像を参照することであって、参照データ構造はグループ化タイプを含む、読み取ることと、記ファイル内の第 1 及び第 2 の画像を読み取ることと、のために構成される回路を含む装置が提供される。

30

【 0 0 3 5 】

本発明の別の態様では、プログラム可能機器に読み込まれ実行されるときに、本発明に記載の方法を実装するための命令のシーケンスを含む、プログラム可能機器のためのコンピュータプログラム製品が提供される。

【 0 0 3 6 】

本発明の別の態様では、本発明に記載の方法を実装するためのコンピュータプログラムの命令を格納するコンピュータ可読記憶媒体が提供される。

40

【 0 0 3 7 】

本発明の別の態様では、実行されると本発明に記載の方法を行わせるコンピュータプログラムが提供される。

【 0 0 3 8 】

本発明の別の態様では、エンティティ（例えば、画像項目又はトラック）を含む符号化されたビットストリームをカプセル化する方法であって、そのカプセル化されたビットストリームはデータ部とメタデータ部を含む、方法が提供される。この方法は、メタデータ部に 1 つ以上のエンティティを識別するエンティティ情報を提供することと、メタデータ部に複数のグループ化情報の中から選択されたグループ化情報、エンティティ

50

情報によって識別される 1 つ以上のエンティティがグループ化情報によって示される所定の基準に基づいてグループを形成することを示すシグナリング、それぞれ異なる所定の基準を示すグループ化情報、を提供することと、ビットストリームを、提供された情報と共にカプセル化されたデータファイルとして出力することと、を含む。

【0039】

一実施形態では、符号化されたビットストリームは、符号化された撮像画像、又は撮像画像を表す符号化されたサンプルを含む。

【0040】

一実施形態では、エンティティ情報は少なくとも 2 つのエンティティを識別する。

10

【0041】

一実施形態では、その方法は、撮像特徴に基づいて撮像装置によって画像を撮像することをさらに含み、所定の基準は撮像ステップの撮像特徴のうちの少なくとも 1 つに関連する。

【0042】

一実施形態では、グループ化情報は 4 文字のコードである。

【0043】

本発明の別の態様では、エンティティ（例えば、画像項目又はトラック）を表す符号化されたビットストリームをカプセル化する方法であって、そのカプセル化されたビットストリームはデータ部及びメタデータ部を含む、方法が提供される。

この方法は、

20

メタデータ部に 1 つ以上のエンティティを識別するエンティティ情報を提供することと、エンティティ情報によって識別される少なくとも 2 つのエンティティが、少なくとも 1 つの共通プロパティに基づいてグループを形成することをシグナリングするグループ化情報をメタデータ部に提供することと、

少なくとも 1 つの共通プロパティについてのプロパティ情報を提供することと、エンティティ情報によって識別される 1 つ以上のエンティティのそれぞれ 1 つを共通プロパティにそれぞれリンクするためのリンク情報を提供することと、ビットストリームを、提供された情報と共にカプセル化されたデータファイルとして出力することと、を含む。

【0044】

30

一実施形態では、エンティティ情報は少なくとも 2 つのエンティティを識別し、リンク情報はエンティティ情報によって識別される少なくとも 2 つのエンティティのそれぞれ 1 つを共通プロパティにリンクする。

【0045】

一実施形態では、プロパティ情報は少なくとも 1 つの共通プロパティ、少なくとも 2 つの異なる値をシグナリングし、リンク情報はエンティティ情報によって識別された少なくとも 2 つのエンティティのそれぞれ 1 つを、共通プロパティの少なくとも 2 つの異なる値にリンクする。

【0046】

一実施形態では、リンク情報はエンティティ情報によって識別される少なくとも 2 つのエンティティのそれぞれ 1 つを、共通プロパティの少なくとも 2 つの異なる値にリンクする。

40

【0047】

一実施形態では、符号化されたビットストリームは、符号化された撮像画像、又は撮像画像を表す符号化されたサンプルを含む。

【0048】

一実施形態では、その方法は、撮像特徴に基づいて撮像装置によって画像を撮像することをさらに含み、共通の特性は撮像ステップの撮像特徴のうちの少なくとも 1 つに関連する。

【0049】

一実施形態では、リンク情報はグループ化構造に含まれるパラメータであり、そのパラメータはプロパティを表すために異なる値をとる。

50

【 0 0 5 0 】

一実施形態では、リンク情報はプロパティ情報をグループに関連付ける構造（例えば、ボックス）である。

【 0 0 5 1 】

本発明の一態様では、ファイル内にメディアデータをカプセル化する方法であって、エンティティのグループを記述するグループ化データ構造を生成することであって、各エンティティはメディアデータの少なくとも一部に対応する、生成することと、エンティティのグループに関連付けられたプロパティを包含するプロパティコンテナデータ構造を生成することと、プロパティとグループ化データ構造との間の関連付け情報を含む関連付けデータ構造を生成することと、プロパティコンテナデータ構造と関連付けデータ構造とを含むプロパティデータ構造を生成することと、グループ化データ構造、プロパティデータ構造、及びメディアデータをファイル内に埋め込むことと、を含む方法が提供される。

10

【 0 0 5 2 】

一実施形態では、この方法は、エンティティのグループに関連付けられた少なくとも1つのテキスト属性と、少なくとも1つのテキスト属性に関連付けられた言語属性と、を含むテキストプロパティを、プロパティコンテナデータ構造内に生成することをさらに含む。

20

【 0 0 5 3 】

一実施形態では、テキストプロパティは、それぞれの複数の言語属性に関連付けられた複数の同じテキスト属性を含む。

【 0 0 5 4 】

一実施形態において、プロパティコンテナデータ構造は第1及び第2のテキストプロパティを含み、第1のテキストプロパティは第1の言語属性に関連付けられたテキスト属性を含み、第2のテキストプロパティは第2の言語属性に関連付けられたテキスト属性を含み、第1の言語は第2の言語とは異なり、関連付けデータ構造は、第1及び第2のテキストプロパティ間の関連付け情報と、グループ化データ構造とを含む。

【 0 0 5 5 】

一実施形態では、テキストプロパティは、名前テキスト属性、記述テキスト属性、及びタグテキスト属性を含む複数のテキスト属性を含む。

30

【 0 0 5 6 】

一実施形態では、この方法は、プロパティコンテナデータ構造内で、名前テキストプロパティ、記述テキストプロパティ、及び/又はタグテキストプロパティのうちの少なくとも1つを生成することであって、テキストプロパティのそれぞれはエンティティのグループに関連付けられたテキスト属性とテキスト属性に関連付けられた言語属性とを少なくとも含む、生成すること、を含む。

【 0 0 5 7 】

一実施形態では、テキストプロパティそれぞれは、複数の言語属性のそれぞれに関連付けられた複数の同じテキスト属性を含む。

40

【 0 0 5 8 】

一実施形態では、プロパティコンテナデータ構造は第1及び第2のテキストプロパティを含み、第1のテキストプロパティは第1の言語属性に関連付けられたテキスト属性を含み、第2のテキストプロパティは第2の言語属性に関連付けられたテキスト属性を含み、第1の言語は第2の言語とは異なり、関連付けデータ構造は、第1及び第2のテキストプロパティ間の関連付け情報と、グループ化データ構造とを含む。

【 0 0 5 9 】

一実施形態では、関連付けデータ構造は、少なくともエンティティとのプロパティの関連付けをさらに含む。

50

【 0 0 6 0 】

本発明の別の態様では、ファイル内にメディアデータをカプセル化する方法であって、エンティティのグループを記述するグループ化データ構造を生成することであって、各エンティティはメディアデータの少なくとも一部に対応する、生成することと、エンティティのグループに関連付けられたテキスト属性とテキスト属性に関連付けられた言語属性とを少なくとも含むグループ化データ構造内にテキストプロパティを生成することと、メディアデータとグループ化データ構造とをファイル内に埋め込むことと、を含む方法が提供される。

【 0 0 6 1 】

一実施形態では、テキストプロパティは、複数の言語属性のそれぞれに関連付けられた複数の同じテキスト属性を含む。

【 0 0 6 2 】

一実施形態では、テキストプロパティは、名前テキスト属性、記述テキスト属性、及びタグテキスト属性を含む複数のテキスト属性を含む。

【 0 0 6 3 】

本発明の別の態様では、ファイル内のメディアデータを読み取る方法であって、エンティティのグループを記述するグループ化データ構造を読み取ることであって、各エンティティはメディアデータの少なくとも一部に対応する、読み取ることと、エンティティのグループに関連付けられたプロパティを包含するプロパティコンテナデータ構造を読み取ることと、プロパティとグループ化データ構造との間の関連付け情報を含む関連付けデータ構造を読み取ることと、プロパティコンテナデータ構造と関連付けデータ構造とを含むプロパティデータ構造を読み取ることと、プロパティに従ってグループ化データ構造内で識別されるメディアデータを読み取ることと、を含む方法が提供される。

【 0 0 6 4 】

一実施形態では、その方法は、エンティティのグループに関連付けられた少なくとも1つのテキスト属性と、少なくとも1つのテキスト属性に関連付けられた言語属性と、を含む、プロパティコンテナデータ構造内のテキストプロパティを読み取ることとをさらに含む。

【 0 0 6 5 】

一実施形態では、テキストプロパティは、複数の言語属性のそれぞれに関連付けられた複数の同じテキスト属性を含む。

【 0 0 6 6 】

一実施形態では、プロパティコンテナデータ構造は第1及び第2のテキストプロパティを含み、第1のテキストプロパティは第1の言語属性に関連付けられたテキスト属性を含み、第2のテキストプロパティは第2の言語属性に関連付けられたテキスト属性を含み、第1の言語は第2の言語とは異なり、関連付けデータ構造は、第1及び第2のテキストプロパティ間の関連付け情報と、グループ化データ構造とを含む。

【 0 0 6 7 】

一実施形態では、テキストプロパティは、名前テキスト属性、記述テキスト属性、及びタグテキスト属性を含む複数のテキスト属性を含む。

【 0 0 6 8 】

一実施形態では、その方法は、プロパティコンテナデータ構造内で、名前テキストプロパティ、記述テキストプロパティ、及び/又はタグテキストプロパティのうちの少なくとも1つを読み取ることであって、前記テキストプロパティのそれぞれはエンティティのグループに関連付けられたテキスト属性と前記テキスト属性に関連付けられた言語属性とを少なくとも含む、読み取ること、

10

20

30

40

50

を含む。

【 0 0 6 9 】

一実施形態では、テキストプロパティそれぞれは、複数の言語属性のそれぞれに関連付けられた複数の同じテキスト属性を含む。

【 0 0 7 0 】

一実施形態では、プロパティコンテナデータ構造は第 1 及び第 2 のテキストプロパティを含み、前記第 1 のテキストプロパティは第 1 の言語属性に関連付けられたテキスト属性を含み、前記第 2 のテキストプロパティは第 2 の言語属性に関連付けられたテキスト属性を含み、前記第 1 の言語は前記第 2 の言語とは異なり、関連付けデータ構造は、前記第 1 及び第 2 のテキストプロパティ間の関連付け情報と、前記グループ化データ構造とを含む。

10

【 0 0 7 1 】

一実施形態では、関連付けデータ構造は、少なくともエンティティとのプロパティの関連付けをさらに含む。

【 0 0 7 2 】

本発明の別の態様では、ファイル内のメディアデータを読み取る方法であって、エンティティのグループを記述するグループ化データ構造を読み取ることであって、各エンティティはメディアデータの少なくとも一部に対応する、読み取ることと、エンティティのグループに関連付けられたテキスト属性と前記テキスト属性に関連付けられた言語属性とを少なくとも含む前記グループ化データ構造内のテキストプロパティを読み取ることと、前記プロパティに従って前記グループ化データ構造内で識別される前記メディアデータを読み取ることと、を含む方法が提供される。

20

【 0 0 7 3 】

一実施形態では、テキストプロパティは、複数の言語属性のそれぞれに関連付けられた複数の同じテキスト属性を含む。

【 0 0 7 4 】

一実施形態では、テキストプロパティは、名前テキスト属性、記述テキスト属性、及びタグテキスト属性を含む複数のテキスト属性を含む。

【 0 0 7 5 】

本発明の別の態様では、ファイル内にメディアデータをカプセル化する装置であって、エンティティのグループを記述するグループ化データ構造を生成することであって、各エンティティはメディアデータの少なくとも一部に対応する、生成することと、エンティティのグループに関連付けられたプロパティを含むプロパティコンテナデータ構造を生成することと、前記プロパティと前記グループ化データ構造との間の関連付け情報を含む関連付けデータ構造を生成することと、前記プロパティコンテナデータ構造と前記関連付けデータ構造とを含むプロパティデータ構造を生成することと、前記グループ化データ構造、前記プロパティデータ構造、及び前記メディアデータを前記ファイル内に埋め込むことと、のために構成される回路を含む装置が提供される。

30

40

【 0 0 7 6 】

本発明の別の態様では、ファイル内にメディアデータをカプセル化する装置であって、エンティティのグループを記述するグループ化データ構造を生成することであって、各エンティティはメディアデータの少なくとも一部に対応する、生成することと、エンティティのグループに関連付けられたテキスト属性と前記テキスト属性に関連付けられた言語属性とを少なくとも含む前記グループ化データ構造内にテキストプロパティを生成することと、前記メディアデータと前記グループ化データ構造とを前記ファイル内に埋め込むことと、のために構成される回路を含む装置が提供される。

【 0 0 7 7 】

50

本発明の別の態様では、ファイル内のメディアデータを読み取る装置であって、
エンティティのグループを記述するグループ化データ構造を読み取ることであって、各エンティティはメディアデータの少なくとも一部に対応する、読み取ることと、
エンティティのグループに関連付けられたプロパティを含むプロパティコンテナデータ構造を読み取ることと、
前記プロパティと前記グループ化データ構造との間の関連付け情報を含む関連付けデータ構造を読み取ることと、
前記プロパティコンテナデータ構造と前記関連付けデータ構造とを含むプロパティデータ構造を読み取ることと、
前記プロパティに従って前記グループ化データ構造内で識別される前記メディアデータを読み取ることと、のために構成される回路を含む装置が提供される。

10

【 0 0 7 8 】

本発明の別の態様では、ファイル内のメディアデータを読み取る装置であって、
エンティティのグループを記述するグループ化データ構造を読み取ることであって、各エンティティはメディアデータの少なくとも一部に対応する、読み取ることと、
エンティティのグループに関連付けられたテキスト属性と前記テキスト属性に関連付けられた言語属性とを少なくとも含む前記グループ化データ構造内のテキストプロパティを読み取ることと、
前記プロパティに従って前記グループ化データ構造内で識別される前記メディアデータを読み取ることと、のために構成される回路を含む装置が提供される。

20

【 0 0 7 9 】

別の実施形態では、プログラム可能機器に読み込まれ実行されるときに、本発明に記載の方法を実装するための命令のシーケンスを含む、プログラム可能機器のためのコンピュータプログラム製品。

【 0 0 8 0 】

別の実施形態では、本発明に記載の方法を実装するためのコンピュータプログラムの命令を格納するコンピュータ可読記憶媒体が提供される。

【 0 0 8 1 】

別の実施形態では、実行されると本発明に記載の方法を行わせるコンピュータプログラムが提供される。

30

【 0 0 8 2 】

本発明による方法の少なくとも一部は、コンピュータで実施することができる。
したがって、本発明は、全体的にハードウェアの実施形態、全体的にソフトウェアの実施形態（ファームウェア、常駐ソフトウェア、マイクロコードなどを含む）、又は本明細書では全て一般に「回路」、「モジュール」、若しくは「システム」と呼ばれることができるソフトウェア及びハードウェアの態様を組み合わせた実施形態の形態をとることができる。さらに、本発明は、媒体に具現化されたコンピュータ使用可能プログラムコードを有する任意の有形の表現媒体に具現化されたコンピュータプログラム製品の形態をとることができる。

【 0 0 8 3 】

40

本発明はソフトウェアで実現することができるため、本発明は、任意の適切なキャリア媒体上のプログラム可能装置に提供するためのコンピュータ可読コードとして実施することができる。有形の非一時的キャリア媒体は、フロッピーディスク、CD-ROM、ハードディスクドライブ、磁気テープ装置又はソリッドステートメモリ装置などの記憶媒体を含んでもよい。過渡搬送媒体は、電気信号、電子信号、光信号、音響信号、磁気信号、又は電磁信号、例えばマイクロ波又はRF信号などの信号を含んでもよい。

【図面の簡単な説明】**【 0 0 8 4 】**

ここで本発明の実施形態は、単なる例として、以下の図面を参照して記載される。

【 0 0 8 5 】

50

【図 1】いくつかの画像又は画像のシーケンスを包含する H E I F ファイルの例を示す。

【 0 0 8 6 】

【図 2】H E I F フォーマットを使用して一連の画像を 1 つのファイルにカプセル化する処理の主なステップを示す。

【 0 0 8 7 】

【図 3】図 2 のカプセル化処理によって生成される H E I F ファイルの解析処理の主なステップを示す。

【 0 0 8 8 】

【図 4】グループ化データ構造を含む、本発明の第 1 の実施形態に係る例を示す。

【 0 0 8 9 】

【図 5】エンティティのグループの範囲内での項目プロパティの項目との関連付けの例を示す。

【 0 0 9 0 】

【図 6】本発明のいくつかの実施形態による、項目プロパティと項目との間の異なる関連付け、エンティティのグループ、又はエンティティのグループの範囲内の項目を示す。

【 0 0 9 1 】

【図 7】本発明のいくつかの実施形態による、H E I F フォーマットを使用して 1 つのファイルに 1 つ以上のエンティティをカプセル化する処理の主なステップを示す。

【 0 0 9 2 】

【図 8】本発明のいくつかの実施形態に従って生成された H E I F ファイルの解析処理の主なステップを示す。

【 0 0 9 3 】

【図 9】本発明の 1 つ以上の実施形態を実施するための計算装置の概略ブロック図である

【発明を実施するための形態】

【 0 0 9 4 】

H E V C 規格は静止画像の符号化のためのプロファイルを定義し、単一の静止画像又は静止画像のバーストを圧縮するための特定のツールを記述する。このような種類の画像データに使用される I S O ベースメディアファイルフォーマット (I S O B M F F) の拡張が、「H E I F 又は High Efficiency Image File Format」という名称で I S O / I E C 2 3 0 0 8 規格の第 1 2 部に含めるために提案されている。

【 0 0 9 5 】

H E I F 及び M I A F 規格は、異なるユースケースに対応する、

- ・デコーダで任意に使用されるタイミングを有し、画像が他の画像に依存し得る画像シーケンスの記憶、及び

- ・単一画像、及び独立して符号化された画像の集合の記憶

の 2 つの形態の記憶をカバーする。

【 0 0 9 6 】

この第 1 のケースでは、カプセル化が I S O ベースメディアファイルフォーマットにおけるビデオトラックのカプセル化に近く (文書「情報技術 - オーディオビジュアルオブジェクトの符号化 - 第 1 2 部： I S O ベースメディアファイルフォーマット」、I S O / I E C 1 4 4 9 6 - 1 2 : 2 0 1 5、第 5 版、2 0 1 5 年 1 2 月を参照)、ファイルレベルが「m o o v」のボックス、「t r a k」のボックス (「m o o v」ボックスにカプセル化されている)、並びにサンプル及びサンプルのグループの記述のためのサンプルグループ化など、同様のツール及び概念が使用される。サンプルは単一の時間に関連付けられた全ての時間設定されたデータ (例えば、ビデオ内のフレーム又は画像シーケンス内の画像) を示す。

【 0 0 9 7 】

ボックスはコンテナとも呼ばれ、ファイル内のデータを記述するために提供されるデータ構造である。ボックスは、一意のタイプ識別子 (通常は 4 文字のコード、F o u r C C 又は 4 C C と記される) と長さによって定義されるオブジェクト指向のビルディングブ

10

20

30

40

50

ロックである。ファイル内の全てのデータ（メディアデータ及びそのメディアデータを記述するメタデータ）は、ボックスに包含されている。このファイル内には他のデータがない。ファイルレベルのボックスは、他のボックスに包含されていないボックスである。

【0098】

「m o o v」ボックスは「t r a k」サブボックスを包含するファイルフォーマットボックスであり、各「t r a k」ボックスはトラック、すなわち関連サンプルの時系列を記述する。

【0099】

この第2のケースでは、I S O B M F F ボックスのセットである「m e t a」ボックスが使用される。これらのボックス及びそれらの階層は「t r a c k r e l a t e d」ボックス（「t r a k」ボックス階層）よりも少ない記述ツールを提供し、関連サンプルの代わりに「情報項目」又は「項目」に関連する。「ボックス」という表現及び「コンテナ」という表現は、両方とも、ファイル内の画像データの構成及び／又はプロパティを記述するメタデータを包含するデータ構造を指すために同じ意味で使用されてもよいことに留意されたい。同じ「ボックス」という表現及び「コンテナ」という表現はファイル中の画像データを包含するメタデータ構造（例えば「m d a t」や「i d a t」ボックス）を指すために同じ意味で使用されてもよい。

【0100】

図1は、1つ以上の静止画像及び可能であればビデオ又は画像のシーケンスのようなメディアデータを包含するH E I Fファイル101の例を示す。このファイルには、ファイルのタイプの識別子（通常は4つの文字コードのセット）を包含する、第1の「f t y p」ボックス（F i l e T y p e B o x）111が包含されている。このファイルには、1つ以上の静止画を記述するメタデータ構造を含む一般的な時間設定のされていないメタデータを包含するために使用される、「m e t a」と呼ばれる第2のボックス（M e t a B o x）102が含まれている。この「m e t a」ボックス102は、いくつかの単一画像を記述する「i i n f」ボックス（I t e m I n f o B o x）121を包含する。各単一画像は、項目1211及び1212とも示されるメタデータ構造I t e m I n f o E n t r y によって記述される。各項目は、一意の32ビット識別子item_IDを有する。これらの項目に対応するメディアデータは、メディアデータ用のコンテナである「m d a t」ボックス104に格納される。

【0101】

任意選択で、画像シーケンス又はビデオの記憶を記述するために、H E I Fファイル101は、いくつかのトラック131及び132を記述する「m o o v」（M o v i e B o x）103と呼ばれる第3のボックスを包含することができる。典型的には、トラック131は時間情報が必ずしも意味がない画像のセットを記述するように設計された画像シーケンス（「p i c t」）トラックであり、トラック122はビデオコンテンツを記述するように設計されたビデオ（「v i d e」）トラックである。これらの両方のトラックは一連の画像サンプルを記述し、画像サンプルは例えばビデオシーケンスのフレームのように、同時に撮像されたピクセルのセットである。2つのトラック間の主な違いは、「p i c t」トラックではタイミング情報は必ずしも意味がないが、「v i d e」トラックについてはタイミング情報がサンプルの表示部のタイミングを制約するように意図されていることである。これらのサンプルに対応するデータは、メディアデータ用のコンテナである「m d a t」ボックス104に格納される。

【0102】

「m d a t」コンテナ104は、データ部分141及び142によって表される項目に対応する時間設定のされていない符号化画像と、データ部分143によって表されるサンプルに対応する時間設定のされた符号化画像とを格納する。

【0103】

H E I Fファイル101の目的は、1つのH E I Fファイルに複数の画像を格納するために利用可能な異なる代替例を記述することである。例えば複数の画像を、項目、又は「p

10

20

30

40

50

「i c t」トラック若しくは「v i d e」トラックであり得るサンプルのトラックのどちらかとして格納できる。実際の選択は、典型的には画像のタイプ及び予期されるファイルの用途に従ってファイルを生成するアプリケーション又は装置によって行われる。

【 0 1 0 4 】

H E I F 標準は画像に関連付けられるプロパティを指定するために設計されたいくつかのメカニズムを、特に、画像のプロパティ、及びより一般的には項目（あらゆる種類のメディアタイプ）のプロパティを宣言又は格納するためのいくつかのメタデータ構造を提供する。典型的には、「m e t a」ボックス 1 0 2 は、任意の項目と順序付けられた一連の項目プロパティとの関連付けを可能にする「i p r p」ボックス（I t e m P r o p e r t i e s B o x）1 2 3 を包含してもよい。この「i p r p」ボックス 1 2 3 は、H E I F ファイルに記述される全ての項目のプロパティを記述する全てのプロパティデータ構造（I t e m P r o p e r t y 及び i t e m F u l l P r o p e r t y）1 2 3 3 を包含するプロパティコンテナデータ構造である「i p c o」ボックス（I t e m P r o p e r t y C o n t a i n e r B o x）1 2 3 1 を包含する。i p r p ボックスは「i p m a」ボックス（I t e m P r o p e r t y A s s o c i a t i o n B o x）のセットも包含しており、これは1つ以上の項目プロパティを所与の項目に実際に関連付けるアソシエーションデータ構造である。その場合、同じプロパティをいくつかの項目に関連付けることが可能である。

10

【 0 1 0 5 】

関連付けられた構文は次のとおりである。

20

```
aligned (8) class ItemProperty(property_type) extends Box(property_type)
{
}
aligned (8) class ItemFullProperty(property_type, version, flags) extends FullBox(property_type, version, flags)
{
}
aligned(8) class ItemPropertyContainerBox extends Box('ipco')
{
Box properties [] ; //ボックスを満たすために
// I t e m P r o p e r t y 又は I t e m F u l l P r o p e r t y から導出されるボックス
}
aligned(8) class ItemPropertyAssociationBox extends FullBox('ipma', version, flags)
{
unsigned int(32) entry_count;
for(i = 0; i < entry_count; i++) {
if (version < 1)
unsigned int(16) item_ID;
else
unsigned int(32) item_ID;
unsigned int(8) association_count;
for (i=0; i < association_count; i++) {
bit(1) essential;
if (flags & 1)
unsigned int(15) property_index;
else
unsigned int(7) property_index;
}
}
```

30

40

50

```

}
}
aligned(8) class ItemPropertiesBox extends Box('iprp') {
ItemPropertyContainerBox property_container;
ItemPropertyAssociationBox association [];
}

```

【0106】

ItemProperty及びItemFullPropertyボックスはプロパティの記述のために設計され、すなわち、全てのプロパティはItemProperty又はItemFullPropertyのいずれかから継承されることになる。ItemPropertyと比較して、ItemFullPropertyは、異なる構文を有する複数のバージョンのプロパティを条件付きでversionパラメータの値に定義でき、オプションの機能又はパラメータを通知/アクティブ化するフラグのマップを条件付きでflagsパラメータの値に定義できる。

10

【0107】

ItemPropertyContainerBoxは、プロパティのセットをItemProperty又はItemFullPropertyボックスの配列として記述するために設計されている。

【0108】

ItemPropertyAssociationBoxは、項目とそのプロパティ間の関連付けを記述するように設計されている。それは項目識別子のリストの説明を提供し、各項目識別情報(item_ID)は、ItemPropertyContainerBox内のプロパティを参照するプロパティインデックスのリストに(1ベースのインデックス値として)関連付けられる。インデックス値0は、プロパティがその項目に関連付けられていないことを示すために予約される。essential属性は、1に設定されている場合には関連付けられたプロパティがその項目に必須であり、そうでない場合には必須ではないことを示す。

20

【0109】

最後に、ItemPropertyContainerBoxとItemPropertyAssociationBoxがItemPropertiesBox内で収集される。

30

【0110】

ISOベースメディアファイルフォーマットは、項目及び/又はトラックのグループ化のために適合されるグループ化機構を明記する。この機構では、「エンティティ」という表現がメディアデータを項目(画像やメタデータ項目などの任意のタイプの項目)又はトラック(ビデオトラック「vide」、画像トラックのシーケンス「pict」、オーディオトラック、又はその他の任意のタイプのトラック)として参照するために使用される。この機構は、エンティティのグループ化を指定する。

【0111】

「meta」ボックス102は、エンティティ1221及び1222のグループを記述するメタデータ構造のセットを包含することができるコンテナボックス「gprl」(GroupsListBox)122を包含してもよい。

40

【0112】

エンティティのグループは、次の構文に従って定義されたEntityToGroupBoxと呼ばれるグループ化データ構造によって記述される。

```

aligned(8) class EntityToGroupBox(grouping_type, version, flags) extends FullBox(grouping_type, version, flags) {
unsigned int(32) group_id;
unsigned int(32) num_entities_in_group;
for(i=0 i num_entities_in_group i++)
unsigned int(32) entity_id;
}

```

50

//残りのデータは特定のグループ化タイプに対して指定されてもよい
}

【0113】

group_idはエンティティのグループの一意的識別子であり、この場合、一意とは、ファイル内で一意であると理解する必要がある。これは、任意のEntityToGroupBoxのgroup_id値、GroupsListBoxを含む階層レベル（ファイル、ムービー、又はトラック）のitem_ID値、又は任意のtrack_ID値（GroupsListBoxがファイルレベルに包含されている場合）と等しくないものとする。次に、entity_idのリストは、グループに属する全てのエンティティを与える。

【0114】

grouping_typeは、グループ化のタイプを指定するために使用される。HEIFは、grouping_type値の制限値を実際に定義する。第1のgrouping_type「alt」は、異なるエンティティが代替的にアプリケーションで使用され得る代替物であることを指定する。第2のgrouping_type「equiv」は、所与の時間設定のされていない画像がトラックのタイムライン内の特有の位置に関連することを指定する。「equiv」エンティティグループに含まれる全ての項目は「均等物」であり、同じ「equiv」エンティティグループ内のトラックは、その項目に対して「均等物」である選択されたサンプルを含む。第3のgrouping_type「ster」は、ステレオ・アプリケーションにおいて、2つのエンティティがステレオ写真、典型的には左右のビューであることを指定する。エンティティの他のグループ化タイプは指定されない。

【0115】

この機構は、指定されたグループのタイプが3つのみであるように、非常に限定されていることに留意されたい。また、グループ上又はグループ内のエンティティ上に、さらなる潜在的な情報又はプロパティをいくつか提供するために指定されているものはない。

【0116】

本発明は、所与の撮像モードに従って撮像された画像のグループを記述するための手段を提供する機構を提供する。これは、使用された撮像モードを記述するための手段を提供する。いくつかの実施形態では、撮像に関するいくつかの追加情報は、画像のグループに関連して記述されてもよい。

【0117】

本発明は、HEIFフォーマットを使用してカプセル化することができるエンティティの他のグループにも適用することができる。例えば、本発明はExifデータのグループなどのメタデータのグループに適用できる。別の例として、本発明はオーディオサンプルのグループに適用することができる。

【0118】

図2は、HEIFフォーマットを使用して一連の画像を1つのファイルにカプセル化する処理の主なステップを示す。カメラは例えば、この処理を適用してもよい。与えられた例は、異なる撮像モードに従ってカメラによって撮像された画像のグループ化に適用される。

【0119】

まず、ステップ201において、一連の写真の撮像モードが決定される。この一連の画像の撮像モードは、シリーズの画像間の関係の種類を記述する。例えば、撮像モードは次のいずれかである。

【0120】

ブラケット撮像モードは、自動露光、ホワイトバランス、フォーカス、フラッシュブラケット撮像モードを含む。これらの全てのブラケットモードは、撮影の1つ以上のパラメータの値が異なる、同じコンテンツのいくつかのショットを行うことで構成される。これらの異なるブラケットモードは、一連の撮像で値が変化するパラメータで異なる。この撮像システムは、その異なるバージョンを生成するために1つの撮像パラメータを変更する。例えば、自動露光ブラケットでは、露光の時間は画像撮像ごとに修正される。

【0121】

10

20

30

40

50

画像バーストは、2つの画像撮像間の短い時間隔で一連の画像を連続的に撮像することにある撮像モードである。

【0122】

パノラマは、複数の画像が各撮像間で重なり合って撮像される撮像モードである。この場合、その原理は、より高い解像度のパノラマを形成するため、撮像された各画像をステッチすることである。

【0123】

タイムラプスは、各ショットの間に予め定められたタイミングで、複数の画像を同じ装置で撮像することにある撮像モードである。

【0124】

ユーザコレクションは、写真シリーズとも呼ばれ、同じコンテキストを共有するシリーズ内の画像をユーザが関連付ける撮像モードである。例えば、フォトグラフは同じ製品のいくつかの写真を作成し、作成した全ての画像を同じファイル内に格納したい。フォトグラフは、セッションの開始時にユーザ収集撮像モードを開始する。フォトグラフは、撮影セッションが終了すると撮影モードを停止する。

【0125】

超解像は、より高い解像度を有する新しい画像を生成するために処理することができる、異なる解像度でいくつかの画像を撮像することにある撮像モードである。

【0126】

多重露光は、画像の多重露光設定の重ね合わせである新しい画像を生成することを目的として、異なる露光でいくつかの画像を撮像することにある撮影モードである。

【0127】

ノイズ低減は、撮像処理によって生成されるランダムノイズを低減するために、単一のシーンのいくつかの画像を撮像することにある撮像モードである。

【0128】

長秒時露光ノイズ低減は、長秒時露光中にセンサ関連のノイズを除去するための撮像モードである。このモードでは、通常の画像に加えて、「ダーク」と呼ばれる画像が、光をセンサに到達させることなく（例えば、レンズ上にキャップを置くことによって、又はシャッターを開かないことによって）、同じ露光持続時間で撮像される。この「ダーク」は、通常の画像からセンサ関連のノイズを除去するために使用できる。

【0129】

口径食補償は、レンズの口径食を補償するための撮像モードである。多くのレンズは光に対して不均一な反応を有する（典型的には、中央よりも隅でレンズを通過してくる光が少ないことから、画像の隅は中央よりも暗くなる）。この不均一性を補償するために、「フラット」と呼ばれる基準画像が、均一に照明された表面の画像を撮ることによって撮像される。この基準画像は、同じレンズでの他の画像撮像の不均一性を補正するために使用することができる。

【0130】

HDR(High Dynamic Range)は、撮像シーンにおける光度の非常に大きな差を処理するための撮像モードである。その結果として得られる画像は、異なる露光を有するいくつかの画像の組合せである。これは自動露光ブラケットモードに類似しているが、より一般的なやり方では画像間の露光変動が規則的でなくてもよく、又は不特定であってもよい。

【0131】

ステップ201で撮像モードが選択されたならば、ステップ202、203、及び204からなる処理ループが適用される。撮像モードの終了まで（例えば、装置のグラフィカルユーザインターフェース又は物理的なユーザインターフェースで特定のオプション又はボタンを起動することによって）、撮像装置はまずステップ203で画像を撮像し、次いでステップ204で、符号化された画像をファイルフォーマットでカプセル化する。撮像ステップ203は、画像の取得と、ビデオ又は静止ピクチャコーデックを使用する際の画像の符号化とを含む。例えば、この例では、コーデックはH.264又はHEVC/H.2

10

20

30

40

50

65フォーマットである。

【0132】

画像の記憶のために、2つの主な可能代替例が利用可能である。

【0133】

第1の代替例では、画像は独立して符号化され、HEIF項目としてファイルに格納される。このカプセル化ステップの間に、撮像の条件に関する追加情報がファイル内に提供されてもよい。例えば、自動露光ブラケットモードの場合、各画像の撮像に使用される露光データが提供されてもよい。この説明は、ItemPropertyボックスのプロパティを使用して提供される。

【0134】

第2の代替例では、画像は「pict」又は「vide」トラックに格納される。追加情報は、SampleEntry又はSampleGroupEntryボックスを使用して提供されてもよい。画像の符号化はビデオ符号化と同様に、HEVCエンコーダを使用して以前の画像に依存してもよい。そのトラック内の以前の画像は、予測符号化のための基準画像として利用可能である。

【0135】

ステップ205において、一連の画像のカプセル化が終了される。このステップの間に、画像のグループとしての撮像された画像の格納は、先に述べた利用可能なグループ化機構の一つを使用してファイル内にシグナリングされる。ステップ204の間に撮像モードのタイプは、グループ化機構内でシグナリングされてもよく、あるいは画像ごとに格納された追加情報内でシグナリングされてもよい。

【0136】

図2で説明したような画像の記憶を行えるようにするために、新しいタイプのグループ化が、関連付けられたプロパティを有するファイル内のこれらのタイプのグループ化をシグナリングするための新しい方法と共に必要である。

【0137】

一実施形態では、2つ以上の撮像モードが一度に使用される。例えば、天文写真では、長秒時露光ノイズ低減と口径食補償の両方が一度に使用される。

【0138】

別の実施形態では、撮像モードが一連の画像から2つ以上のグループを生成する。例えば、タイムラプスでは、最初のグループ化は全ての画像を包含し、一方で2番目のグループ化は1つおきの画像を包含する。

【0139】

別の実施形態では、ステップ205に続いて、撮像編集ステップが実現される。この撮像編集ステップでは、新しい画像のグループが作成される。この撮像編集ステップは複数回実現されることができ、その結果、いくつかの画像のグループの追加が起こる。

【0140】

最初の可能なことは、初期の撮像モードと同じ撮像モードに対応する新しいグループ化を追加することである。例えば、撮影時には、毎秒の画像を包含するタイムラプスが実現される。タイムラプス画像は、撮像の終わりに一緒にグループ化される。撮像編集時に、1秒おきの画像を有する別のタイムラプスに対応する、2番目のグループ化が作成される。

【0141】

別の可能なことは、別の撮像モードに対応する新しいグループ化を追加することである。例えば、撮像時には、自動露光ブラケットグループが作成される。次いで撮像編集時に、自動露光ブラケットグループのサブセットを包含するHDRグループ化が作成される。このHDRグループ化は、HDR画像を生成するための自動露光ブラケットグループ化の画像をいくつか選択した結果である。

【0142】

[撮像モードごとの1つのグループ化タイプ]

第1の実施形態によれば、グループ化タイプは、各撮像モードに対して定義される。この

10

20

30

40

50

グループ化タイプは、グループ化情報内に記述される。

【0143】

図4は、この第1の実施形態による例を示す。

【0144】

ItemInfoBox 403には、画像が項目404、405、及び406として記述されている。

【0145】

グループ化情報ボックス、例えばEntityToGroupBox 401は、撮像モードに基づくグループ化タイプを含む。次に、グループ化情報ボックス401は、そのグループに属する項目の項目識別子のリスト402を含む。

10

【0146】

例えば、汎用的な「brak」グループ化タイプがブラケット撮像モードのために定義されてもよい（ここでの「brak」コードは一例である）。特有のタイプのブラケットモード、即ち、自動露光ブラケット、ホワイトバランスブラケット、フォーカスブラケット、フラッシュ露光ブラケット、被写界深度ブラケット及びISOブラケットは、「brak」グループ化タイプのパラメータを用いてシグナリングされてもよい。例えば、エンティティのグループ化機構では、EntityToGroupがブラケットのタイプを決定するbracketing_typeパラメータを包含する。通常、bracketing_typeパラメータは、予約済みの4文字コードである。4文字コードの1つの値は、各ブラケットタイプごとに予め定められている。別の例では、サンプルのグループ化機構が使用され、「brak」グループ化タイプのパラメータがVisualSampleGroupEntryの特定の種類で記述されている（VisualSampleGroupEntryから継承している）。

20

【0147】

あるいは、特定のグループ化タイプは、各特有のブラケット撮像モードごとに定義されてもよい。「aebr」は自動露光ブラケット撮影に対して、「wbb r」はホワイトバランスブラケット撮影に対して、「fobr」はフォーカスブラケット撮影に対して、「afbr」はフラッシュ露光ブラケット撮影に対して、「dob r」は被写界深度ブラケット撮影に対して、「isbr」はISOブラケット撮影に対して定義されてもよい。

【0148】

30

タイムラプス撮像モードについて、グループ化タイプ「tila」が定義されてもよい。このタイプのグループ化は、写真がいくつか撮られる、時間が重要である全ての用途をカバーする。主な用途は、写真間の規則的な時間隔の有無に関わらず、タイムラプスに対してである。タイムラプスのパラメータ（例えば、そのタイムラプスの合計持続時間、各画像間のオフセット）はグループ化のパラメータを使用して（例えば、EntityToGroup又はVisualSampleGroupEntryパラメータとして）シグナリングされてもよい。

【0149】

グループ化タイプ「pano」は、パノラマグループ化タイプに対して定義されてもよい。これは、写真がいくつか撮られる、空間次元が重要である全ての用途を含む。これは、パノラマだけでなく所与の項目の異なるビューも含む。パノラマのパラメータ（例えば、左から右、右から左、zスキャン又はスパイラルスキャンなどの撮像パターン）はグループ化のパラメータを使用して（例えば、EntityToGroup又はVisualSampleGroupEntryパラメータとして）シグナリングされてもよい。

40

【0150】

グループ化タイプ「imbu」又は「brst」は、画像バーストグループ化タイプに対して定義されてもよい。これは、いくつかの画像が途切れることなく連続的に撮像される全ての場合をカバーする。画像バーストのパラメータ（例えば、各撮像間の時間オフセット、及び撮像の合計持続時間）はグループ化のパラメータを使用して（例えば、EntityToGroup又はVisualSampleGroupEntryパラメータとし

50

て)シグナリングされてもよい。

【0151】

グループ化タイプ「ucol」、「pser」、「labl」、又は「udcs」は、ユーザコレクション、同じタグ若しくはラベルを有する写真シリーズ又は写真のセット、又はユーザ定義の撮像シリーズ撮像モードに対して定義されてもよい。例えば自動処理は、同じ場所に、及び/又は予め定められた時間隔内に、写真ショットをグループ化してもよい。例えば撮像装置は、撮像画像の(例えば、GPSセンサからの)位置情報を使用して、その位置に対応する町の名前を決定する。同じ町で撮られた全ての画像は、画像の1つのグループを形成する。ユーザコレクションのパラメータ(例えば、撮像の位置又はコンテンツの説明)はグループ化のパラメータを使用して(例えばEntityToGroupやVisualSampleGroupEntryパラメータとして)シグナリングされてもよい。

10

【0152】

「sres」グループ化タイプは、超解像撮像モードに対して定義されてもよい。これは、より高い解像度を有する新しい画像を生成することを意図していくつかの画像が連続的に撮像される全ての場合をカバーする。超解像グループのパラメータ(例えば、新しい画像の意図された解像度)はグループ化のパラメータを使用して(例えば、EntityToGroup又はVisualSampleGroupEntryパラメータとして)シグナリングされてもよい。

【0153】

「mult」グループ化タイプは、多重露光撮像モードに対して定義されてもよい。これは、グループ内の画像の重ね合わせによって形成される新しい画像を生成することを意図して、いくつかの画像が連続的に撮像される全ての場合をカバーする。多重露光のパラメータ(例えば、グループの各画像に対するアルファブレンディング係数を示すことができる、画像のセットをブレンドするためのパラメータ)はグループ化のパラメータを使用して(例えば、EntityToGroup又はVisualSampleGroupEntryパラメータとして)シグナリングされてもよい。

20

【0154】

グループ化タイプ「nois」は、ノイズ低減撮像モードに対して定義されてもよい。これは、より少ないノイズを有する新しい画像を生成することを意図していくつかの画像が連続的に撮像される全ての場合をカバーする。

30

【0155】

グループ化タイプ「dark」は、長秒時露光ノイズ低減撮像モードのために定義されてもよい。これは、1つ以上の「ダーク」な画像(すなわち、光をセンサに到達させずに撮像された画像)と並んで1つ以上の画像が撮像される全ての場合をカバーする。長秒時露光ノイズ低減のパラメータ(例えば、どの画像がダークな画像に対応するか)はグループ化のパラメータを使用して(例えば、EntityToGroup又はVisualSampleGroupEntryパラメータとして)シグナリングされてもよい。

【0156】

グループ化タイプ「flat」は、口径食補償撮像モードに対して定義されてもよい。これは、1つ以上の「フラット」な画像(すなわち、均一に照明されたシーン上で撮像された画像)と並んで1つ以上の画像が撮像される全ての場合をカバーする。口径食補償のパラメータ(例えば、どの画像がフラットな画像に対応するか)はグループ化のパラメータを使用して(例えば、EntityToGroup又はVisualSampleGroupEntryパラメータとして)シグナリングされてもよい。

40

【0157】

グループ化タイプ「hdr」は、HDR撮像モードに対して定義されてもよい。これは、HDR画像を生成するために1つ以上の画像が撮像される全ての場合をカバーする。HDR撮像モードのパラメータ(例えば、最も明るい領域に対応するものから最も暗い領域に対応するものまでの画像の順序付け)はグループ化のパラメータを使用して(例えば、

50

EntityToGroup又はVisualSampleGroupEntryパラメータとして)選ばれてもよい。

【0158】

他のグループ化タイプは、これらのグループ化タイプの2つ以上を組み合わせで定義されてもよい。例えば、天文画像の場合、グループ化タイプ「astr」は、長秒時露光ノイズ低減と口径食補償との両方を組み合わせで定義されてもよい。

【0159】

例えば、フォトグラフはAEブラケットを使用して2つの写真を作る。その結果、3×2のサンプルが得られる。各写真は、1つの画像項目に埋め込まれる。

【0160】

したがって、HEIFファイルは以下で説明する6つの項目を包含する。

- ・項目1:item_id=1 //短い露光時間(例えば、exposure_stop=-2)で撮像される
- ・項目2:item_id=2 //中程度の露光時間(例えば、exposure_stop=0)で撮像される
- ・項目3:item_id=3 //長い露光時間(例えば、exposure_stop=2)で撮像される
- ・項目4:item_id=4 //短い露光時間(exposure_stop=-2)で撮像される
- ・項目5:item_id=5 //中程度の露光時間(例えば、exposure_stop=0)で撮像される
- ・項目6:item_id=6 //長い露光時間(例えば、exposure_stop=2)で撮像される

【0161】

グループ化タイプ値「aebr」を持つEntityToGroupBoxが定義される。2つのエントリ(3つのサンプルの各AEシリーズに対して1つ)が存在する。

```
Goups ListBox{
EntityToGroupBox('aebr') {
group_id='1234' //第1のブラケットセットの識別子
num_entities_in_group=3 //ブラケットセットサイズが3に等しい
entity_id=1 //第1の項目参照
entity_id=2 //第2の項目参照
entity_id=3 //第3の項目参照
}
EntityToGroupBox('aebr') {
group_id='1235' //第2のブラケットセットの識別子
num_entities_in_group=3 //ブラケットセットサイズが3に等しい
entity_id=4 //第2のセットの第1の項目、すなわち第4の項目参照
entity_id=5 //第3のセットの第1の項目、すなわち第5の項目参照
entity_id=6 //第4のセットの第1の項目、すなわち第6の項目参照
}
}
```

【0162】

[汎用撮像シリーズグループ化タイプ]

第2の実施形態では、「撮像シリーズ」のための汎用グループ化タイプ「case」が定義される(4文字コード「case」は1つの例であり、任意の未使用の予約された4文字コードが使用されてもよい)。このグループ化タイプは、先に述べた全ての撮像モードに対して使用するよう意図されている。シリーズの画像が項目として記述されるとき、このグループ化情報は、「case」に等しいGrouping_typeパラメータを有するEntityToGroup情報ボックスである。シリーズの画像がトラック(「vide」又は「pict」トラックのいずれか)でのサンプルとして記述されている場合、そのグループ化情報は、例えば「case」に等しいGrouping_typeを有するSampleToGroup及びSampleGroupDescriptionBox情報ボックスである。

【0163】

HEIFが異なる撮像モードを有するいくつかの画像を包含する場合、それらは全て同じ「case」グループに所属する。一連の項目又はサンプルが撮像シリーズグループに所

10

20

30

40

50

属する場合、カプセル化は、グループ化タイプ「case」を指定しなければならない。異なる撮像間を区別するために、撮像モードのタイプがプロパティ情報でシグナリングされてもよい。例えば、SampleEntry、SampleGroupDescriptionEntry、ItemProperty、又はItemFullPropertyが、そのグループのいくつかの項目のシリーズに関連する撮像モード及び/又はパラメータを正確にするために使用されてもよい。

【0164】

例えば、画像が項目として格納されている場合、ItemPropertyContainerBoxは、一連の画像の撮像モードを記述する一連のItemPropertyボックスを包含する。例えば、あるプロパティは自動露光ブラケット撮影モードを記述し、一方で別のプロパティは撮像に使用された実際の露光値を記述する。

10

【0165】

別の実施形態では、プロパティ情報がVisualSampleEntry情報として指定される。この実施形態では、SampleToChunkBoxはカメラに記録されたマルチ画像ファイルのmoov/trak/stblボックス階層に存在する。チャンクは、サンプルの連続したセットである。SampleToChunkBoxのsample_description_indexは、このチャンクのサンプルに適用されるSampleEntryの「stsd」ボックス内のインデックスを提供する。特定のビジュアルサンプルエントリは通常のビジュアルサンプルエントリ（例えば、ISO/IEC 14496-15におけるような、「hvc1」又は「avc1」タイプ）と比較して、余分なボックスを包含して定義される。この余分なボックスはグループ化タイプに対するパラメータ、例えば、ブラケットモードの記述、又はユーザによって提供されるタグ若しくはラベル（又は本発明の非限定的なリストによって示唆されるような任意の特定のプロパティ）を提供する。

20

【0166】

さらに別の実施形態では、画像のグループ化は2つ以上の撮像モードを組み合わせる。この実施形態では、grouping_type「case」を使用してグループ化が指定される。次に、各撮像モードは、それ自身のプロパティ情報を使用して指定される。

【0167】

自動露光ブラケット撮影モードに関して、第1の実施形態では、露光の値は、ItemProperty又はVisualSampleEntryとして、プロパティ情報内で次のように記述される。

30

1. Syntax for Item Property

```
aligned (8) class AutoExposureBracketingProperty extends ItemProperty('aeb
r') {
int (8) exposure_stop;
}
```

2. Syntax for Sample Entry

Box type : 'aeb'

Container : VisualSampleEntry

Mandatory : No

Quantity : Zero or one

```
aligned (8) class AutoExposureBracketing extends Box ('aeb') {
```

```
int (8) exposure_stop;
```

3. Syntax for Sample Group Entry

```
aligned (8) class AutoExposureBracketingEntry extends VisualSampleGroupEn
try('aeb') {
```

```
int (8) exposure_stop
```

4. Semantic

・ exposure_stopはこの項目のEV絞りを指定する整数値である。

【0168】

40

50

別の実施形態によれば、各画像の露光値は、プロパティ情報、典型的には `ItemProperty` で指定された情報から計算される。例えば、構文は次のようになる。

```
aligned (8) class AutoExposureBracketingProperty extends ItemProperty('aeb
r') {
int (8) exposure_step; /*full=1, half=2, thind=3, quarter=4*/
int (8) exposure_stop;
}
```

【0169】

ここで、`exposure_step`は、露光ブラケットに使用される増分ステップを指定する整数値である。例えば、1に等しい場合は絞りインクリメントがそのまま使用され、2に等しい場合は1/2絞りインクリメントが使用され、3に等しい場合は三番目の絞りインクリメントが使用され、4に等しい場合は1/4絞りインクリメントが使用される。`exposure_stop`パラメータは、最大絞り値である。

【0170】

結果として、グループ化情報における宣言順の最初の画像は、露光時間が最も短い画像となる。このグループ化情報の画像数は奇数である。この画像の露光絞りは、式、 $-1 \times \text{exposure_stop} / \text{exposure_step}$ の結果に等しい。

【0171】

その最後の画像は、`exposure_stop / exposure_step`に等しい露光絞りを有する。グループ内の画像の数を2で割ったものに等しいインデックスを有する画像（すなわち、中央位置の画像）は、0に等しい露光絞りを有する。画像数が3より多い場合、`exposure_stop`の間隔 $[-\text{exposure_stop} / \text{exposure_step}, \text{exposure_stop} / \text{exposure_step}]$ は、そのグループに記述される全ての画像に均等に分配される。

【0172】

さらに別の実施形態によれば、ブラケットセット内の画像の順序（グループ化情報、すなわち、`EntityToGroup`又は`SampleToGroup`ボックス内の記述の順序）は、自然な順序（昇順）とは異なる。このプロパティ情報は、ブラケットセットの順序を記述する新しいパラメータを含む。この構文は次のようであってもよい。

```
aligned (8) class AutoExposureBracketingProperty extends ItemProperty('aeb
r') {
int (8) exposure_order;
int (8) exposure_step;
int (8) exposure_stop;
}
```

【0173】

新たな`exposure_order`パラメータは、ブラケットセット内の画像の順序を指定する。

`exposure_order`パラメータが0に等しい場合は、この順序付けは、`exposure_stop`間隔内の`exposure_stop`の昇順による。例えば、この順序は $\{-2 \text{ EV}, 0 \text{ EV}, 2 \text{ EV}\}$ である。

`exposure_order`パラメータが1に等しい場合は、この順序付けは、`exposure_stop`間隔内の`exposure_stop`の降順による。例えば、この順序は $\{2 \text{ EV}, 0 \text{ EV}, -2 \text{ EV}\}$ である。

`exposure_order`パラメータが2に等しい場合は、この順序付けは、まず0 EVに等しい`exposure_stop`の画像によって始まり、昇順で他の画像が続く。例えば、この順序は $\{0 \text{ EV}, -2 \text{ EV}, 2 \text{ EV}\}$ である。

`exposure_order`パラメータが3に等しい場合は、この順序付けは、まず0 EVに等しい`exposure_stop`の画像によって始まり、降順で他の画像が続く。例えば、この順序は $\{0 \text{ EV}, 2 \text{ EV}, -2 \text{ EV}\}$ である。

【0174】

ホワイトバランスブラケットの場合、プロパティ情報は、`ItemProperty`、`I`

10

20

30

40

50

`temFullProperty`、又は`SampleEntry`若しくは`SampleGroupEntry`のいずれかにおける画像の色温度をシグナリングする。この構文は次のようであってもよい。

1. Syntax for Item Entry

```
aligned (8) class WhiteBalanceBracketingProperty extends ItemProperty('wbbr') {
    unsigned int (16) color_temperature
}
```

2. Syntax for Sample Entry

Box type : 'wbbr' ' 10

Container : VisualSampleEntry

Mandatory : No

Quantity : Zero or one

```
aligned (8) class WhiteBalanceBracketingEntry extends Box('wbbr') {
    unsigned int (32) color_temperature
}
```

3. Syntax for Sample Group Entry

```
aligned (8) class WhiteBalanceBracketing extends VisualSampleGroupEntry('wbbr') {
    unsigned int (32) color_temperature
}
```

20

4. Semantic

`color_temperature`は、例えばケルビン単位の色温度値を表す。別の代替例では、この色温度が`color_temperature`から計算される。例えば、パラメータの各値は、予め定められた色温度に対応する。例えば、ケルビン単位の色温度は、 $\text{color_temperature} \times 10 + 2200$ に等しい。

【0175】

代替りの実施形態では、ホワイトバランスブラケットプロパティ情報（「wbbr」）が、その関連付けられた画像に初期カメラ設定と比較して適用される、青／黄色軸及び／又はマゼンタ／緑軸上のホワイトバランス補償を示す。

30

【0176】

`VisualSampleEntry`及び`SampleGroupEntry`のボックスの構文は同じパラメータを包含し、同じ4文字コードを定義する。この差は、`VisualSampleEntry`の場合、新しいクラスがボックスクラスを拡張し、1つの`VisualSampleEntry`コンテナで定義されることになることである。`SampleGroupEntry`の場合、新しいクラスは、項目プロパティに対して定義されているのと同じ4文字コードを用いて`VisualSampleGroupEntry`ボックスを拡張する。

【0177】

この構文は以下であってもよい。

40

1. Syntax for ItemProperty

```
aligned (8) class FocusBracketingItemProperty extends ItemProperty('fobn') {
    int (8) blue_amber;
    int (8) green_magenta;
}
```

2. Semantic

`blue_amber`欄は、「色温度」を表す。

`Green_magenta`欄は「色合い」を表す。

【0178】

関連付けられた画像項目を参照するエンティティグループ「wbbr」が1つ及び1つの

50

み存在するはずである。プロパティ情報が `SampleGroupEntry` の場合、それぞれ異なる `grouping_type_parameter` の値を有する、「w b b r」タイプのサンプルグループがいくつか存在してもよい。

【0179】

フォーカスブラケットの場合、プロパティ情報は、`ItemProperty`、`ItemFullProperty`、又は `SampleEntry` 若しくは `SampleGroupEntry` のいずれかで画像を撮像するときに使用されるフォーカス距離をシグナリングする。この構文は次のようであってもよい。

1. Syntax for ItemProperty

```
aligned (8) class FocusBracketingItemProperty extends ItemProperty('fobr') {
    unsigned int (32) focus_distance_numerator;
    unsigned int (32) focus_distance_denominator;
```

10

2. Syntax for Sample Entry

`Box type : 'fobn'`

`Container : VisualSampleEntry`

`Mandatory : No`

`Quantity : Zero or one`

```
aligned (8) class FocusBracketing extends Box ('fobr') {
    unsigned int (32) focus_distance_numerator
    unsigned int (32) focus_distance_denominator
}
```

20

3. Semantic

フォーカス距離は例えば、`focus_distance_numerator` 欄と `focus_distance_denominator` 欄との比率としてメートル単位で表される。別の実施形態では、これらの2つのパラメータがミリメートル単位で距離を表す `focus_distance` パラメータによって置き換えられる。

【0180】

画像バースト撮像モードの場合、プロパティ情報は、`ItemProperty`、`ItemFullProperty`、又は `SampleEntry` 若しくは `SampleGroupEntry` ボックスのいずれかで撮像のインデックスをシグナリングする。以下は、項目プロパティとしてのプロパティ情報のあり得る構文である。`VisualSampleEntry` 及び `SampleGroupEntry` のボックスのこの構文は同じパラメータを包含し、同じ4文字コードを定義する。この差は、`VisualSampleEntry` の場合、新しいクラスがボックスクラスを拡張し、1つの `VisualSampleEntry` コンテナで定義されることになることである。`SampleGroupEntry` の場合、新しいクラスは、項目プロパティに対して定義されているのと同じ、コードされた4文字を用いて `VisualSampleGroupEntry` ボックスを拡張する。

30

【0181】

第1の実施形態では、この構文は以下のようであり得る。

40

```
aligned(8) class ImageBurstProperty extends ItemProperty('imbu') {
    unsigned int (32) burst_index
}
```

【0182】

これはこのセマンティックを有する。

・ `burst_index` は、画像のバースト内の項目のインデックスを表す。

【0183】

第2の実施形態では、バースト内の画像のインデックスがその項目の宣言の順序から推論される。これは記述するプロパティの数を削減し、HEIFファイルがよりコンパクトになる。以下は、項目プロパティとしてのプロパティ情報の構文である。`VisualSa`

50

`SampleEntry` 及び `SampleGroupEntry` のボックスの構文は同じパラメータを包含し、同じ4文字コードを定義する。この差は、`VisualSampleEntry` の場合、新しいクラスがボックスクラスを拡張し、1つの `VisualSampleEntry` コンテナで定義されることになることである。`SampleGroupEntry` の場合、新しいクラスは、項目プロパティに対して定義されているのと同じ4文字コードを用いて `VisualSampleGroupEntry` ボックスを拡張する。

【0184】

この構文は以下であってもよい。

```
aligned(8) class ImageBurstProperty extends ItemProperty('imbu') {
  unsigned int (1) infer_index_value;
  if (infer_index_value == 0)
    unsigned int (32) burst_index
}
```

10

【0185】

これは以下のセマンティクスを有する。

- ・ `infer_index_value` は、画像のバースト内の項目のインデックスが「`i i n f`」ボックス内の項目の位置インデックスと等しいことを示すブール値である。
- ・ `burst_index` は、画像のバースト内の項目のインデックスを表す。

【0186】

別の実施形態では、`infer_index_value` は画像のバースト内の項目のインデックスがグループ化情報内の項目のインデックスと等しいことを示す。この実施形態は2つのグループを定義することによって、いくつかの順序付け（例えば、正規順序及び逆順序）を有利に定義することを可能にする。

20

【0187】

別の実施形態では、画像バーストプロパティ情報が、バーストの2つの画像間の時間隔を、ミリ秒単位で表される符号なし整数値として含む。

【0188】

この構文は以下であってもよい。

```
aligned (8) class ImageBurstProperty extends ItemProperty('imbu') {
  unsigned int (32) time_delta;
}
```

30

【0189】

これは以下のセマンティクスを有する。

- ・ `time_delta` は、画像項目とバースト内の以前の項目との間に経過した100分の1秒の数である。
- ・ このプロパティは、タイプ「`b r s t`」のエンティティグループ内の最初の画像項目には存在しないか、又は無視される必要がある。

【0190】

さらに別の実施形態では、画像バーストプロパティ情報が画像のバースト内の関連付けられた画像の撮像時間を定義する。好ましくは、時間ゼロが撮像装置によって定義される基準時間に対応する。これは、複数の画像バーストが同じ基準時間を使用し、複数の画像バーストが同じ画像項目を含むように定義することを可能にする。この構文は以下であってもよい。

40

```
aligned (8) class ImageBurstProperty extends ItemProperty('imbu') {
  unsigned int (32) capture_time;
}
```

【0191】

これは以下のセマンティクスを有する。

- ・ `capture_time` は、画像項目と撮像装置からの基準時間の間に経過した100分の1秒の数である。

50

【 0 1 9 2 】

さらに別の実施形態では、画像バーストプロパティ情報は、そのプロパティがどのバーストの画像に適用されるかを定義する。これにより、画像間の間隔が異なるいくつかのバーストに画像項目を所属させることができる。例えば、第1のバーストは毎秒撮像された画像を包含することができ、一方、第2のバーストは、1秒おきに撮像された画像のみを包含する。

【 0 1 9 3 】

この構文は以下であってもよい。

```
aligned(8) class ImageBurstProperty extends ItemProperty('imbu') {
  unsigned int(32) time_delta;
  unsigned int (32) group_id;
}
```

10

【 0 1 9 4 】

これは以下のセマンティクスを有する。

- ・time_deltaは、画像項目とバースト内の以前の項目との間に経過した100分の1秒の数である。

- ・group_idは、ターゲット化されたEntityToGroupの識別子である。

【 0 1 9 5 】

このプロパティは、タイプ「burst」のエンティティグループ内の最初の画像項目には存在しないか、又は無視される必要がある。

20

【 0 1 9 6 】

他の実施形態では、画像バースト撮像モードのためのプロパティ情報が「burst」4文字コードを使用する。

【 0 1 9 7 】

タイムラプス撮像モードの場合、プロパティ情報は、連続する画像の撮像間の間隔、又は画像の撮像時間をシグナリングする。このプロパティ情報は、「tila」4文字コードを使用できる。これは、時間関連のパラメータが使用されるImageBurstPropertyと同じ構文及びセマンティクスを有することができる。

【 0 1 9 8 】

画像シリーズ又はユーザコレクション又はプロパティ情報は、ItemProperty、ItemFullProperty、又はSampleEntry若しくはSampleGroupEntryボックスのいずれかのセット内の撮像された画像のインデックスをシグナリングする。以下は、項目プロパティとしてのプロパティ情報の構文である。VisualSampleEntry及びSampleGroupEntryのボックスの構文は同じパラメータを包含し、同じ4文字コードを定義する。この差は、VisualSampleEntryの場合、新しいクラスがボックスクラスを拡張し、1つのVisualSampleEntryコンテナで定義されることになることである。SampleGroupEntryの場合、新しいクラスは、項目プロパティに対して定義されているのと同じ4文字コードを用いてVisualSampleGroupEntryボックスを拡張する。

30

40

【 0 1 9 9 】

この構文は以下であってもよい。

```
aligned(8) class PhotoSeriesProperty extends ItemProperty('pser') {
  unsigned int (32) series_index
}
```

【 0 2 0 0 】

これは以下のセマンティクスを有する。

- ・series_indexは、画像のバースト内の項目のインデックスを表す。

【 0 2 0 1 】

別の実施形態では、写真シリーズのインデックスが画像バーストプロパティ情報に対して

50

行われたように推論され、したがって、記述バイトを節約する。

【0202】

フラッシュ露光ブラケット情報は、ItemPropertyボックス、ItemFullPropertyボックス、又はSampleEntryボックス若しくはSampleGroupEntryボックスのいずれかで、撮像画像のフラッシュ露光値をシグナリングする。以下は、項目プロパティとしてのプロパティ情報の構文である。VisualSampleEntry及びSampleGroupEntryのボックスの構文は同じパラメータを包含し、同じ4文字コードを定義する。この差は、VisualSampleEntryの場合、新しいクラスがボックスクラスを拡張し、1つのVisualSampleEntryコンテナで定義されることになることである。SampleGroupEntryの場合、新しいクラスは、項目プロパティに対して定義されているのと同じ4文字コードを用いてVisualSampleGroupEntryボックスを拡張する。

10

【0203】

この構文は以下であってもよい。

```
aligned(8) class Flash ExposureBracketingProperty extends ItemProperty('afb
r') {
int (8) flash_exposure_numerator;
int (8) flash_exposure_denominator;
}
```

20

【0204】

これは以下のセマンティクスを有する。

flash_exposure_numerator及びflash_exposure_denominatorは、flash_exposure_numeratorとflash_exposure_denominatorとの比率として、f絞りで表される項目又はサンプルのフラッシュ露光値を表す整数である。

【0205】

被写界深度ブラケット情報は、ItemPropertyボックス、ItemFullPropertyボックス、又はSampleEntryボックス若しくはSampleGroupEntryボックスのいずれかにおいて、撮像された画像の口径変化値をシグナリングする。以下は、項目プロパティとしてのプロパティ情報の構文である。VisualSampleEntry及びSampleGroupEntryのボックスの構文は同じパラメータを包含し、同じ4文字コードを定義する。この差は、VisualSampleEntryの場合、新しいクラスがボックスクラスを拡張し、1つのVisualSampleEntryコンテナで定義されることになることである。SampleGroupEntryの場合、新しいクラスは、項目プロパティに対して定義されているのと同じ4文字コードを用いてVisualSampleGroupEntryボックスを拡張する。

30

【0206】

この構文は以下であってもよい。

```
aligned(8) class DepthOfFieldBracketingProperty extends ItemProperty('dobr') {
int (8) f_stop_numerator;
int (8) f_stop_denominator;
}
```

40

【0207】

これは以下のセマンティクスを有する。

ここで、口径変化は、整数値であるf_stop_numeratorとf_stop_denominatorとの比率として、f絞りの数で表される。

【0208】

パノラマ情報は、画像がパノラマ撮像モードを使用して撮像された画像のグループに所属することをシグナリングする。この情報は、ItemProperty、ItemFull

50

lProperty、SampleEntry、又はSampleGroupEntryのいずれかにある。以下は、項目プロパティとしてのプロパティ情報の構文である。VisualSampleEntry及びSampleGroupEntryのボックスの構文は同じパラメータを包含し、同じ4文字コードを定義する。この差は、VisualSampleEntryの場合、新しいクラスがボックスクラスを拡張し、1つのVisualSampleEntryコンテナで定義されることになることである。SampleGroupEntryの場合、新しいクラスは、項目プロパティに対して定義されているのと同じ4文字コードを用いてVisualSampleGroupEntryボックスを拡張する。

【0209】

10

この構文は以下であってもよい。

```
aligned(8) class PanoramaBracketingProperty extends ItemProperty('pano') {
    unsigned int (1) panorama_direction;
    unsigned int (15) frame_number;
}
```

【0210】

これは以下のセマンティクスを有する。

panorama_directionはパノラマ方向をシグナリングする1ビットのフラグ(0:左から右、1:右から左)であり、frame_numberはパノラマ方向に従うフレーム番号を(低い方から高い方へと)表す整数である。

20

【0211】

別の実施形態では、この構文は以下であってもよい。

```
aligned(8) class PanoramaBracketingProperty extends ItemProperty('pano') {
    unsigned int (1) panorama_direction;
    unsigned int (7) reserved;
}
```

【0212】

これは以下のセマンティクスを有する。

panorama_directionはパノラマ方向をシグナリングする1ビットフラグ(0:左から右、1:右から左)である。このフレームは、デフォルトのグループ順序付けに従って順序付けられる。

30

【0213】

別の実施形態では、この構文は以下であってもよい。

```
aligned(8) class PanoramaBracketingProperty extends ItemProperty('pano') {
    unsigned int (8) panorama_type;
    if (panorama_type == 4) {
        unsigned int (8) rows;
        unsigned int (8) columns;
    }
    if (panorama_type == 5) {
        unsigned int (8) rows;
        for (i = 0; i < rows; i++) {
            unsigned int (8) columns;
        }
    }
    if (panorama_type == 8) {
        unsigned int (32) panorama_subtype;
    }
}
```

40

【0214】

50

これは以下のセマンティクスを有する。

panorama_typeは符号なしの整数、例えば8ビットであり、使用されるパノラマのタイプを以下のように通知する。

- ・ 0：左から右への水平パノラマ
- ・ 1：右から左への水平パノラマ
- ・ 2：上から下への垂直パノラマ
- ・ 3：下から上への垂直パノラマ
- ・ 4：2つの追加パラメータを有する格子パノラマ

o 行は格子の行の数である。

o 列は格子内の行の数である。このパラメータは存在しない可能性があり、列の数はグループ内の画像の総数及び行数から計算される。

- ・ 5：他のパラメータをいくつか有する格子パノラマ

o 行は格子の行の数である。

o 列は格子内の各列に対する行の数である。

・ 6：前、左、後、右、上、下の順序に従って順序付けられた画像を有する立方体パノラマ。他の順序付けを、又は異なる順序付けを有する他のタイプの立方体パノラマを定義できる可能性がある。

- ・ 7：そのパノラマの正確なタイプが指定されていない、未指定のパノラマ。

・ 8：パノラマのタイプを識別する4CCである別のパラメータpanorama_subtypeを有する、ユーザ定義のパノラマ。

【0215】

水平パノラマ及び垂直パノラマについては、そのパノラマが最後の画像が最初の画像と結合される完全な360°パノラマであるかどうかを示すためにフラグを追加することができる。

【0216】

グリッドパノラマ（規則的であるか否かにかかわらず）については、そのパノラマが水平方向、垂直方向、又は両方向に360°をカバーするかどうかを示すために、同様にフラグを追加することができる。

【0217】

超解像情報は、その画像が超解像撮像モードを使用して撮像された画像のグループに所属することをシグナリングする。この情報は、ItemProperty、ItemFullProperty、SampleEntry、又はSampleGroupEntryのいずれかにある。下は、項目プロパティとしてのプロパティ情報の構文である。VisualSampleEntry及びSampleGroupEntryのボックスの構文は同じパラメータを包含し、同じ4文字コードを定義する。この差は、VisualSampleEntryの場合、新しいクラスがボックスクラスを拡張し、1つのVisualSampleEntryコンテナで定義されることになることである。SampleGroupEntryの場合、新しいクラスは、項目プロパティに対して定義されているのと同じ4文字コードを用いてVisualSampleGroupEntryボックスを拡張する。

【0218】

この構文は以下であってもよい。

```
aligned(8) class SuperResolutionProperty extends ItemProperty('sres') {
}
```

【0219】

多重露光情報は、その画像が多重露光撮像モードを使用して撮像された画像のグループに所属することをシグナリングする。この情報は、ItemProperty、ItemFullProperty、SampleEntry、又はSampleGroupEntryのいずれかにある。下は、項目プロパティとしてのプロパティ情報の構文である。VisualSampleEntry及びSampleGroupEntryのボックスの

10

20

30

40

50

構文は同じパラメータを包含し、同じ4文字コードを定義する。この差は、VisualSampleEntryの場合、新しいクラスがボックスクラスを拡張し、1つのVisualSampleEntryコンテナで定義されることになることである。SampleGroupEntryの場合、新しいクラスは、項目プロパティに対して定義されているのと同じ4文字コードを用いてVisualSampleGroupEntryボックスを拡張する。

【0220】

この構文は以下であってもよい。

```
aligned(8) class MultiExposureProperty extends ItemProperty('mult') {
  unsigned int (8) weight;
```

10

【0221】

これは以下のセマンティクスを有する。

・weightは、この画像のアルファブレンディング係数を示す。

【0222】

ノイズ低減情報は、その画像がノイズ低減撮像モードを使用して撮像された画像のグループに所属することをシグナリングする。この情報は、ItemProperty、ItemFullProperty、SampleEntry、又はSampleGroupEntryのいずれかにある。下は、項目プロパティとしてのプロパティ情報の構文である。VisualSampleEntry及びSampleGroupEntryのボックスの構文は同じパラメータを包含し、同じ4文字コードを定義する。この差は、VisualSampleEntryの場合、新しいクラスがボックスクラスを拡張し、1つのVisualSampleEntryコンテナで定義されることになることである。SampleGroupEntryの場合、新しいクラスは、項目プロパティに対して定義されているのと同じ4文字コードを用いてVisualSampleGroupEntryボックスを拡張する。

20

【0223】

この構文は以下であってもよい。

```
aligned(8) class NoiseReductionProperty extends ItemProperty('nois') {
}
```

30

【0224】

長秒時露光ノイズ低減情報は、画像がItemProperty、ItemFullProperty、SampleEntry、又はSampleGroupEntryのいずれかの暗部に対応することをシグナリングする。下は、項目プロパティとしてのプロパティ情報の構文である。VisualSampleEntry及びSampleGroupEntryのボックスの構文は同じパラメータを包含し、同じ4文字コードを定義する。この差は、VisualSampleEntryの場合、新しいクラスがボックスクラスを拡張し、1つのVisualSampleEntryコンテナで定義されることになることである。SampleGroupEntryの場合、新しいクラスは、項目プロパティに対して定義されているのと同じ4文字コードを用いてVisualSampleGroupEntryボックスを拡張する。

40

【0225】

この構文は以下であってもよい。

```
aligned(8) class LongExposureNoiseReductionProperty extends ItemProperty('
dark') {
}
```

【0226】

これは以下のセマンティクスを有する。

LongExposureNoiseReductionPropertyを有する項目は、「ダーク」な画像に対応する。

50

【0227】

口径食補償情報は、画像が `ItemProperty`、`ItemFullProperty`、`SampleEntry`、又は `SampleGroupEntry` のいずれかのフラットに対応することをシグナリングする。下は、項目プロパティとしてのプロパティ情報の構文である。`VisualSampleEntry` 及び `SampleGroupEntry` のボックスの構文は同じパラメータを包含し、同じ4文字コードを定義する。この差は、`VisualSampleEntry` の場合、新しいクラスがボックスクラスを拡張し、1つの `VisualSampleEntry` コンテナで定義されることになることである。`SampleGroupEntry` の場合、新しいクラスは、項目プロパティに対して定義されているのと同じ4文字コードを用いて `VisualSampleGroupEntry` ボックスを拡張する。

10

【0228】

この構文は以下であってもよい。

```
aligned (8) class VignettingCompensationProperty extends ItemProperty('flat') {
}
```

【0229】

これは以下のセマンティクスを有する。

`VignettingCompensationProperty` を有する項目は、「フラット」な画像に対応する。

【0230】

HDR情報は、`ItemProperty`、`ItemFullProperty`、`SampleEntry`、又は `SampleGroupEntry` のいずれかにおける画像の順序付けをシグナリングする。下は、項目プロパティとしてのプロパティ情報の構文である。`VisualSampleEntry` 及び `SampleGroupEntry` のボックスの構文は同じパラメータを包含し、同じ4文字コードを定義する。この差は、`VisualSampleEntry` の場合、新しいクラスがボックスクラスを拡張し、1つの `VisualSampleEntry` コンテナで定義されることになることである。`SampleGroupEntry` の場合、新しいクラスは、項目プロパティに対して定義されているのと同じ4文字コードを用いて `VisualSampleGroupEntry` ボックスを拡張する。

20

【0231】

この構文は以下であってもよい。

```
aligned (8) class HDRProperty extends ItemProperty('hdr') {
  unsigned int (8) item_order;
}
```

30

【0232】

これは以下のセマンティクスを有する。

`item_order` は、そのグループ内の項目の順序付けである。

【0233】

代替例として、HDR情報は、その画像がHDR撮像モードを使用して撮像された画像のグループに所属することのみをシグナリングする。この情報は、`ItemProperty`、`ItemFullProperty`、`SampleEntry`、又は `SampleGroupEntry` のいずれかにある。下は、項目プロパティとしてのプロパティ情報の構文である。`VisualSampleEntry` 及び `SampleGroupEntry` のボックスの構文は同じパラメータを包含し、同じ4文字コードを定義する。この差は、`VisualSampleEntry` の場合、新しいクラスがボックスクラスを拡張し、1つの `VisualSampleEntry` コンテナで定義されることになることである。`SampleGroupEntry` の場合、新しいクラスは、項目プロパティに対して定義されているのと同じ4文字コードを用いて `VisualSampleGroupEntry` ボックスを拡張する。

40

【0234】

50

この構文は以下であってもよい。

```
aligned(8) class HDRProperty extends ItemProperty('hdr') {
}
```

【 0 2 3 5 】

これら全てのプロパティについて、適用可能な場合には、そのパラメータ値は絶対又は相対のいずれかになり得る。絶対値は、その基準がグループの外部にあることを意味する。相対値は、その値がグループの別の項目の値とその値の差分として指定されることを意味する。デフォルトでは、グループ化の最初の項目の場合に、相対値に使用される基準項目。場合によっては、基準項目がグループ化内の以前の項目になる可能性がある。例えば、タイムラプスにおける撮像時間は、予め定義された基準時間に対応する絶対時間であるか、又は以前の画像との時間差のみが与えられる相対時間である。

10

【 0 2 3 6 】

好ましくは、ブラケット処理のために、撮像設定はカメラ設定に関連して指定される。これは、同じブラケットモードを使用するいくつかのグループ化が、プロパティ情報からの値に対して同じ基準を共有することを可能とする。このように、同じ撮像モードを使用するいくつかのグループに所属する項目には、この撮像モードに対応するプロパティ情報が1つだけ添付される。

【 0 2 3 7 】

一実施形態では、相対値を使用する場合、プロパティ情報はそのグループの識別子を包含する。これは、各グループのプロパティ情報が、それが包含している識別子によってそのグループにリンクされるため、同じ撮像モードを使用して項目をいくつかのグループに所属させることを可能にする。

20

【 0 2 3 8 】

本実施形態では、自動露光ブラケット撮影モードに対するプロパティ情報は以下であってもよい。

```
aligned(8) class AutoExposureBracketingProperty extends ItemProperty('aebr') {
int (8) exposure_stop;
unsigned int(32) group_id;
}
```

【 0 2 3 9 】

ここで、group_id は、このプロパティに関連付けられ、プロパティが適用される項目を所有するグループの識別子である。

30

【 0 2 4 0 】

SampleToGroupに関して、デフォルトでは、サンプルは単一のグループにのみ関連付けることができる。しかし、SampleToGroupボックスのgrouping_type_parameterを使用すると、同じサンプルに対していくつかのグループを定義できる。したがって、好ましい実施形態では、本発明で定義されるグループ化タイプのうちの1つに従ってサンプルをグループ化するとき、grouping_type_parameter欄は、単一のサンプルをいくつかのグループにグループ化することを可能にするために使用されてもよい。

【 0 2 4 1 】

以下の例では、自動露光ブラケット撮影モードを使用して2つの異なる撮影が行われている。各撮像の結果、3つの異なる画像が撮像される。2回の撮像の結果は6つの画像になる。各画像は1つの項目に埋め込まれる。

40

【 0 2 4 2 】

ファイル構造は次のようになるだろう。

- ・ 項目 1 : ItemId=1; exposure_stop= - 2 // exposure=low
- ・ 項目 2 : ItemId=2; exposure_stop=0 // exposure=mid
- ・ 項目 3 : ItemId=3; exposure_stop=2 // exposure=high
- ・ 項目 4 : ItemId=4; exposure_stop= - 1 // exposure=low
- ・ 項目 5 : ItemId=5; exposure_stop=0 // exposure=mid

50

・項目 6 : ItemId=6 ; exposure_stop= - 1 // exposure=high

【 0 2 4 3 】

「 c a s e 」グループ化タイプ値を有する E n t i t y T o G r o u p B o x が定義される。2つのエントリ (3つのサンプルの各 A E シリーズに対して1つ) が存在する。

```
GoupsListBox{
EntityToGroupBox('case') {
group_id='1234' //第1のブラケットセット
num_entities_in_group=3
entity_id= 1 //項目 # 1 を参照
entity_id= 2 //項目 # 2 を参照
entity_id= 3 //項目 # 3 を参照
}
```

10

```
EntityToGroupBox('case'){
group_id='1235' //第2のブラケットセット
num_entities_in_group= 3
entity_id= 4 //項目 # 4 を参照
entity_id= 5 //項目 # 5 を参照
entity_id= 6 //項目 # 6 を参照
}
```

20

```
ItemPropertyContainerBox { //全ての項目で使用されている全てのブラケットプロパティを一覧表示
```

```
AutoExposureBracketingProperty('aebr'){ // property_index= 1
exposure_stop = - 2
}
```

```
AutoExposureBracketingProperty('aebr'){ // property_index= 2
exposure_stop = - 1
}
```

```
AutoExposureBracketingProperty('aebr'){ // property_index= 3
exposure_stop = 0
}
```

30

```
AutoExposureBracketingProperty('aebr'){ // property_index= 4
exposure_stop = 1
}
```

```
AutoExposureBracketingProperty('aebr'){ // property_index= 5
exposure_stop = 2
}
```

```
ItemPropertyAssociation { //露光entry_countを記述するプロパティに各項目を関連付ける
```

40

```
entry_count = 6 // 6 項目
```

```
//項目 1
```

```
item_id = 1
```

```
association_count = 1
```

```
essential = 0 //このプロパティは必須ではない
```

```
property_index = 1 //property_index= 1 のプロパティの露光
```

```
//項目 2
```

```
item_id = 2
```

```
association_count = 1
```

```
essential = 0
```

50

```

property_index = 3 //property_index = 3 のプロパティの露光
//項目 3
item_id = 3
association_count = 1
essentital = 0
property_index = 5 //property_index = 5 のプロパティの露光
//項目 4
item_id = 4
association_count = 1
essentaial = 0
property_index = 2 //property_index = 2 のプロパティの露光
//項目 5
item_id = 5
association_count = 1
essentaial = 0
property_index = 3 //property_index = 3 のプロパティの露光
//項目 6
item_id = 6
association_count = 1
essential = 0
property_index = 4 //property_index = 4 のプロパティの露光
【 0 2 4 4 】

```

10

20

別の実施形態では、異なる露光レベルの一連のいくつかの画像、例えば3つの画像にあるショットの画像をグループ化するようにカメラが構成される。この例では、カメラは *Visual Sample Entry* のプロパティ情報、すなわちトラックの1つ以上のサンプルを記述するメタデータを指定する。結果として得られるマルチ画像ファイルでは、撮影中に撮られた画像の数に等しい数のサンプル、この例では3つを包含することになる。例に従って、新しい特定の *Visual Sample Entry* と3つのサンプルの対応するチャンクが、録画されたマルチ画像ファイルの「*stbl*」ボックスに作成される。この新しいショットの *SampleToChunkBox* の *sample_description_index* は、「*stbl*」ボックス内の特定の *Visual Sample Entry* のインデックスに等しく設定される。その撮影シーケンスの終わりにユーザは、全てのピクチャを包含し記述する1つのトラックから構成されるマルチ画像ファイルを得る。カメラUIを通して、ユーザは、ショットのシーケンスを一連の3つの画像として見ることができる。

30

【 0 2 4 5 】

別の例では、異なるやり方で編成された画像が、露光レベルごとにアクセスされてもよい。カメラが露光レベルごとに画像をグループ化するように構成されている場合、そこで各チャンクは1つの画像を包含する。露光レベル毎に1つの特定の視覚サンプルエントリが宣言される。N個の特定の *Visual Sample Entries* が宣言され、カメラが常に同じ順序（例えば、露光不足、通常、露光過多）でピクチャを撮像する場合、*SampleToChunkBox* の *sample_description_index* は法N（ショット内のピクチャインデックス）に等しく設定することができる。撮影シーケンスの終わりにユーザは、全てのピクチャを包含し記述する1つのトラックから構成されるマルチ画像ファイルを得る。カメラUIを介して、記録されたピクチャは、それらの露光レベルに従って編成されて見られてもよい。露光レベル当たりのサンプルを説明するために、以前の実施形態で説明したような特定のサンプル群の使用、特にサンプルのパターンに依拠する *Compact Sample Group* の使用が好ましい。ここでは、それは露光レベルが同じ順序で発生する場合である。パターンが識別されてサンプルグループがサンプルのパターンとして定義され、各サンプルグループは、対応する *SampleGroupDescriptionBox* 内のサンプルグループエントリに関連付けられる。古典的なサンプルグル

40

50

ープ、すなわちパターンのないサンプルグループの使用も可能であり、サンプルエントリの変更はいくつかの実装においてデコーダのリセットにつながる可能性があり、これはサンプルグループを用いた場合ではないため、Visual Sample Entryよりも好ましい。

【0246】

[プロパティに関連付けられた汎用撮像シリーズグループ化タイプ]

この第3の実施形態は、汎用撮像シリーズグループ化タイプに関する第2の実施形態の変形例である。

【0247】

この変形例では、グループ内の全ての項目に共通のパラメータ（例えば、露光、ホワイトバランス、パノラマなど）を提供し、グループ内の項目に特有のパラメータを正確にする項目プロパティとグループをリンクするために、汎用撮像時間グループ化タイプに対して追加のパラメータ（例えば、撮像シリーズに対して「case」と記される）が定義される。特に、「capture_mode」パラメータは、項目に特有のパラメータ（例えば、画像項目として記述された画像）をさらに記述するプロパティ情報（プロパティのproperty_type）を識別する4文字コードをシグナリングする。その原理は、1つのHEIFパーサが汎用のcapture-timeグループ化タイプの「capture_mode」パラメータからプロパティ情報のタイプ（つまり、ItemProperty又はSampleGroupDescriptionBox又はSampleEntryのタイプ）を決定できることである。

【0248】

その原理は、1つのHEIFリーダーが所与の項目のグループに関連付けられた全ての項目プロパティの間で項目プロパティを容易に検索できることである。

【0249】

EntityToGroupとして記述されている場合のグループ化情報の構文は、以下に記述する。

```
aligned(8) class EntityToGroupBox('case', version, flags) extends
FullBox('case', version, flags) {
    unsigned int(32) group_id;
    unsigned int(32) num_entities_in_group;
    for(i=0 i num_entities_in_group i++)
        unsigned int(32) entity_id; //以下のパラメータは、grouping_type「case」に
        対する共通のパラメータを提供する
        unsigned int(32) capture_mode; //撮像モードを識別する4CC
    }
```

【0250】

ここで、capture_modeは撮像モードを定義する。同じ4文字コードを有する項目プロパティは、所与のcapture_modeの項目特有のパラメータを提供する。

【0251】

グループ内の項目ごとに、「capture_mode」と同じ4CCを有する項目プロパティが、この「capture_mode」の項目特有のパラメータを定義する。

【0252】

例：capture_modeに対して、以下の4CCコードが定義できる。

10

20

30

40

【表 1】

撮像モードタイプ	4ccコード
自動露光ブラケット	aebr
ホワイトバランスブラケット	wbbr
フォーカスブラケット	fobr
フラッシュ露光ブラケット	afbr
被写界深度ブラケット	dobr
パノラマ	pano
ユーザ定義の撮像シリーズ	udcs
タイムラプス	tila
超解像	sres
多重露光	mult
ノイズ低減	nois
長秒時露光ノイズ低減	dark
口径食補償	flat
HDR	hdr

10

20

上記のcapture_modeに定義する項目プロパティの例。

【 0 2 5 3 】

```
aligned(8) class AutoExposureBracketingProperty extends ItemProperty('aebr') {
int (8) exposure_step;
int (8) exposure_numerator;
}
```

30

【 0 2 5 4 】

ここで、exposure_stepは、露光ブラケットに使用される増分ステップを指定する整数値である。1に等しい場合は絞りインクリメントがそのまま使用され、2に等しい場合は1/2絞りインクリメントが使用され、3に等しい場合は三番目の絞りインクリメントが使用され、4に等しい場合は1/4絞りインクリメントが使用される。exposure_numeratorは、項目の露光値絞りを計算するために使用される露光分子を指定する整数値である。

【 0 2 5 5 】

ここで、関連付けられた項目の露光値 (EV) 絞りは、以下の式の結果に等しくなる。

$$\text{exposure_numerator} / \text{exposure_step}$$

40

【 0 2 5 6 】

```
aligned (8) class WhiteBalanceBracketingProperty extends ItemProperty('wbbr') {
unsigned int (32) color_temperature
}
```

【 0 2 5 7 】

ここで、color_temperatureは、色温度値をケルビン単位で表す。

【 0 2 5 8 】

50

別の実施形態では、WhiteBalanceBracketingPropertyが以下のように定義される。

```
aligned (8) class WhiteBalanceBracketingProperty extends ItemProperty('wbb
r') {
int (8) blue_amber;
int (8) green_magenta;
}
```

【 0 2 5 9 】

ここで、blue_amberは、青 - 黄色軸上のホワイトバランス補償の値である。このblue_amber欄は、「色温度」を表す。

10

【 0 2 6 0 】

そして、green_magentaは、緑 - マゼンタ軸上のホワイトバランス補償の値である。このgreen_magenta欄は「色合い」を表す。

【 0 2 6 1 】

```
aligned (8) class FocusBracketingProperty extends ItemProperty('fobr') {
unsigned int (32) focus_distance_numerator
unsigned int (32) focus_distance_denominator
}
```

【 0 2 6 2 】

ここで、フォーカス距離は、focus_distance_numeratorとfocus_distance_denominatorの比としてメートル単位で表される。

20

【 0 2 6 3 】

```
aligned (8) class FlashExposureBracketingProperty extends ItemProperty('afb
r') {
int (8) flash_exposure_numerator;
int (8) flash_exposure_denominator;
}
```

【 0 2 6 4 】

ここで、フラッシュ露光補正はflash_exposure_numeratorとflash_exposure_denominatorとの比としてf絞りの数で表される。

30

【 0 2 6 5 】

```
aligned (8) class DepthOfFieldBracketingProperty extends ItemProperty('dobr') {
int (8) f_stop_numerator;
int (8) f_stop_denominator;
}
```

【 0 2 6 6 】

ここで、口径変化は、f_stop_numeratorとf_stop_denominatorとの比として、f絞りの数で表される。

【 0 2 6 7 】

```
aligned(8) class PanoramaBracketingProperty extends ItemProperty('pano') {
unsigned int (1) panorama_direction
unsigned int (15) frame_number;
}
```

40

【 0 2 6 8 】

ここで、panorama_directionはパノラマ方向をシグナリングする1ビットフラグ(0:左から右、1:右から左)であり、frame_numberはパノラマ方向に従ったフレーム番号を(低い方から高い方へと)表す整数である。

【 0 2 6 9 】

別の実施形態では、この構文は以下であってもよい。

```
aligned(8) class PanoramaBracketingProperty extends ItemProperty('pano') {
```

50

```

unsigned int (1) panorama_direction;
unsigned int (7) reserved;
}

```

【 0 2 7 0 】

これは以下のセマンティクスを有する。

panorama_directionはパノラマ方向をシグナリングする1ビットフラグ(0:左から右、1:右から左)である。このフレームは、デフォルトのグループ順序付けに従って順序付けられる。

【 0 2 7 1 】

別の実施形態では、この構文は以下のようであり得る。

```

aligned(8) class PanoramaBracketingProperty extends ItemProperty('pano') {
  unsigned int (8) panorama_type;
  if (panorama_type == 4) {
    unsigned int (8) rows;
    unsigned int (8) columns;
  }
  if (panorama_type == 5) {
    unsigned int (8) rows;
    for (i = 0; i < rows; i++) {
      unsigned int (8) columns;
    }
  }
  if (panorama_type == 8) {
    unsigned int (32) panorama_subtype;
  }
}

```

【 0 2 7 2 】

これは以下のセマンティクスを有する。

panorama_typeは符号なしの整数、例えば8ビットであり、使用されるパノラマのタイプを以下のように通知する。

- ・ 0 : 左から右への水平パノラマ
- ・ 1 : 右から左への水平パノラマ
- ・ 2 : 上から下への垂直パノラマ
- ・ 3 : 下から上への垂直パノラマ
- ・ 4 : 2つの追加パラメータを有する格子パノラマ

○ 行は格子の行の数である。

○ 列は格子内の行の数である。

このパラメータは存在しなくてもよい可能性があり、列の数はグループ内の画像の総数及び行数から計算される。

- ・ 5 : 他のパラメータをいくつか有する格子パノラマ

○ 行は格子の行の数である。

○ 列は格子内の各列に対する行の数である。

・ 6 : 前、左、後、右、上、下の順序に従って順序付けられた画像を有する立方体パノラマ。

他の順序付けを、又は異なる順序付けを有する他のタイプの立方体パノラマを定義できる可能性がある。

- ・ 7 : そのパノラマの正確なタイプが指定されていない、未指定のパノラマ。

・ 8 : パノラマのタイプを識別する4ビットである別のパラメータpanorama_subtypeを有する、ユーザ定義のパノラマ。

【 0 2 7 3 】

10

20

30

40

50

水平パノラマ及び垂直パノラマについては、そのパノラマが最後の画像が最初の画像と結合される完全な360°パノラマであるかどうかを示すためにフラグを追加することができる。

【0274】

グリッドパノラマ（規則的であるか否かにかかわらず）については、そのパノラマが水平方向、垂直方向、又は両方向に360°をカバーするかどうかを示すために、同様にフラグを追加することができる。

【0275】

```
aligned(8) class UserDefinedCaptureSeriesProperty extends ItemProperty('ud
cs') {
    utf8string label
}
```

10

【0276】

ここで、labelは、UTF-8文字のnull終端文字列で、ユーザ定義の撮像シリーズについて人間に可読な名前を提供する。

【0277】

タイムラプスプロパティの構文は次のようになる。

```
aligned(8) class TimeLapseProperty extends ItemProperty('tila') {
    unsigned int(32) time_delta;
}
```

20

【0278】

これは以下のセマンティクスを有する。

- ・time_deltaは、画像項目とバースト内の以前の項目との間に経過した100分の1秒の数である。全ての項目が規則的な時間フレーム上で撮像されるときに、2つの連続する項目間の経過時間を使用することにより、全ての項目に対して単一のプロパティを使用することが可能になる。

- ・このプロパティは、タイプ「tila」のエンティティグループ内の最初の画像項目には存在しないか、又は無視される必要がある。

【0279】

さらに別の実施形態では、タイムラプスプロパティ構文が以下のようであり得る。

30

```
aligned(8) class TimeLapseProperty extends ItemProperty('tila') {
    unsigned int(32) capture_time;
}
```

【0280】

これは以下のセマンティクスを有する。

- ・capture_timeは、画像項目と撮像装置からの基準時間の間に経過した100分の1秒の数である。外部基準に関連する値を使用することにより、2つの項目が、それらに同じ撮像時間を与えることによって同時に撮像された（すなわち、同期された）ことをシグナリングすることが可能になる。

【0281】

40

別の実施形態では、この構文は以下のようであり得る。

```
aligned(8) class TimeLapseProperty extends ItemProperty('tila') {
    unsigned int(32) time_delta;
    unsigned int(32) group_id;
}
```

【0282】

これは以下のセマンティクスを有する。

- ・time_deltaは、画像項目とバースト内の以前の項目との間に経過した100分の1秒の数である。

- ・group_idは、ターゲット化されたEntityToGroupの識別子である。

50

【 0 2 8 3 】

このプロパティは、タイプ「t i l a」のエンティティグループ内の最初の画像項目には存在しないか、又は無視される必要がある。

【 0 2 8 4 】

```
aligned(8) class MultiExposureProperty extends ItemProperty('mult') {
  unsigned int (8) weight;
}
```

【 0 2 8 5 】

これは以下のセマンティクスを有する。

・weightは、この画像のアルファブレンディング係数を示す。

10

【 0 2 8 6 】

```
aligned(8) class LongExposureNoiseReductionProperty extends ItemProperty('
dark') {
}
```

【 0 2 8 7 】

これは以下のセマンティクスを有する。

LongExposureNoiseReductionPropertyを有する項目は、「ダーク」な画像に対応する。

【 0 2 8 8 】

```
aligned(8) class VignettingCompensationProperty extends ItemProperty('flat') {
}
```

20

【 0 2 8 9 】

これは以下のセマンティクスを有する。

VignettingCompensationPropertyを有する項目は、「フラット」な画像に対応する。

【 0 2 9 0 】

これら全てのプロパティについて、適用可能な場合には、そのパラメータ値は絶対又は相対のいずれかになり得る。絶対値は、その基準がグループの外部にあることを意味する。相対値は、その値がグループの別の画像の値とその値との差分として指定されることを意味する。デフォルトでは、グループ化の最初の項目の場合に、相対値に使用される基準項目。場合によっては、基準項目がグループ化内の以前の項目になることがある。例えば、タイムラプスにおける撮像時間は、予め定義された基準時間に対応する絶対時間であるか、又は以前の画像との時間差のみが与えられる相対時間である。

30

【 0 2 9 1 】

好ましくは、ブラケット処理のために、撮像設定はカメラ設定に関連して指定される。これは、同じブラケットモードを使用するいくつかのグループ化が、プロパティ情報からの値に対して同じ基準を共有することを可能とする。このように、同じ撮像モードを使用するいくつかのグループに所属する項目には、この撮像モードに対応するプロパティ情報が1つだけ添付される。

【 0 2 9 2 】

一実施形態では、相対値を使用する場合、プロパティ情報はそのグループの識別子を包含する。これは、各グループのプロパティ情報が、それが包含している識別子によってそのグループにリンクされるため、同じ撮像モードを使用して項目をいくつかのグループに所属させることを可能にする。

40

【 0 2 9 3 】

本実施形態では、自動露光ブラケット撮影モードに対するプロパティ情報は以下であってもよい。

```
aligned (8) class AutoExposureBracketingProperty extends ItemProperty('aeb
r') {
  int (8) exposure_step;
  int (8) exposure_numerator;
```

50

```
unsigned int (32) group_id;
}
```

【 0 2 9 4 】

ここで、group_idは、このプロパティに関連付けられ、プロパティが適用される項目を所有するグループの識別子である。

【 0 2 9 5 】

3つの画像が項目として格納される、自動露光ブラケットモードを使用するこの実施形態の例。

【 0 2 9 6 】

ファイル構造は次のようになってもよい。

10

- ・項目 # 1 : Itemid=1 exposure_stop=-2 // exposure=low
- ・項目 # 2 : Itemid=2 exposure_stop=0 // exposure=mid
- ・項目 # 3 : Itemid=3 exposure_stop=2 // exposure=high

【 0 2 9 7 】

「case」グループ化タイプ値と「aebr」撮像モードを有するEntityToGroupBoxが以下に定義されている。

```
GroupsListBox{
```

```
EntityToGroupBox('case') {
```

```
group_id='1234'
```

```
num_entities_in_group=3
```

20

```
entity_id=1
```

```
entity_id=2
```

```
entity_id=3
```

```
capture_mode='aebr'
```

```
}
```

```
}
```

```
ItemPropertyContainerBox {
```

```
AutoExposureBracketingProperty('aebn'){
```

```
exposure_step = 1;
```

```
exposure_numerator = -2;
```

30

```
}
```

```
AutoExposureBracketingProperty('aebn'){
```

```
exposure_step = 1 ;
```

```
exposure_numerator = 0;
```

```
}
```

```
AutoExposureBracketingProperty('aebn'){
```

```
exposure_step = 1 ;
```

```
exposure_numerator = 2
```

```
}
```

```
}
```

40

```
ItemPropertyAssociation {
```

```
entry_count = 3
```

```
item_id = 1
```

```
association_count = 1
```

```
essential = 0
```

```
property_index = 1
```

```
item_id = 2
```

```
association_count = 1
```

```
essential = 0
```

```
property_index = 2
```

50

```

item_id = 3
association count = 1
essential = 0
property_index = 3
}

```

【0298】

画像のグループは、1つ以上の「p i c t」トラックにカプセル化することもできる。各画像は同期サンプル（全てイントラ）とすることができ、又はインター予測を使用してもよい。

【0299】

例えば、画像シーケンスの場合、その原理はサンプルのグループ化に依拠し、撮像モードの種類ごとに専用のgrouping_type及びサンプルグループの記述を定義することである。例えば、「a e b r」は、自動露光ブラケットに対する画像サンプルのグループを示す。対応するV i s u a l S a m p l e G r o u p E n t r yは次のように定義される。

```

aligned (8)class AutoExposureBracketingEntry extends VisualSampleGroupEntry('aebr'){
int (8) exposure_order;
int (8) exposure_step;
int (8) exposure_stop;
}

```

【0300】

ここで、exposure_orderパラメータは、ブラケットセット内の画像の順序を指定する。exposure_stepは、露光ブラケットに使用される増分ステップを指定する整数値である。1に等しい場合は絞りインクリメントがそのまま使用され、2に等しい場合は1 / 2絞りインクリメントが使用され、3に等しい場合は三番目の絞りインクリメントが使用され、4に等しい場合は1 / 4絞りインクリメントが使用される。exposure_stopはサンプルの最大露光値(E V)絞りを指定する整数値である。

【0301】

その場合、関連付けられた画像の露光絞りは、以下の式の結果に等しくなる。

$$- 1 \times \text{exposure_stop} / \text{exposure_step}$$

【0302】

他の全ての撮像モードに対しても同様の定義を定義できる。

【0303】

撮像シリーズに関連するプロパティが経時的に変化しない場合、デフォルトのサンプルグループ化機構を使用することができ、又は項目プロパティをV i s u a l S a m p l e E n t r yで定義することができる（「p a s p」、「c o i r」、又は「c l a p」項目プロパティなど）。

【0304】

この実施形態の代替例では、撮像モードは、専有のボックスに記述されたP r o p e r t y I n f o r m a t i o nを指す。結果として、capture_modeパラメータは、専有のボックス（例えば、「u u i d」ボックス）の4文字コードを指す。このような場合、グループ化情報は、その専有情報が「c a s e」グループ内の画像に属するのかを正確に識別するデータをさらに記述する。例えば、capture_modeが「u u i d」と等しい場合、そのプロパティ情報は「u u i d」タイプのボックスのユーザタイプ欄を指定する。H E I Fリーダは、撮像モードをさらに記述する専有ボックスを有利に決定することができる。この実施形態は、異なる専有ボックスを区別することを可能にする4文字コード（一般にs u b _ t y p e又はproprietary_typeとして参照される）を含む任意の他の専有ボックス拡張システムにも適用される。

【0305】

この実施形態の別の代替例では、第1の実施形態で列挙されたグループ化タイプがさらに

10

20

30

40

50

、そのグループ化タイプと等しい4文字コードを有するプロパティ情報を指定してもよい。例えば撮像装置は、「a e b r」に等しいgrouping_typeを有する自動露光ブラケットグループを宣言し、そのグループに属する各項目にI t e m P r o p e r t i e s「a e b r」を関連付ける。

【0306】

例えば、画像項目の場合、原則として、異なるグループ化目的（撮像モード）に対応するE n t i t y T o G r o u pに対するいくつかの特定のグループ化タイプを定義する。同じ4 C Cは、所与のE n t i t y T o G r o u pのgrouping_typeと、対応するグループ内の項目に特有のパラメータを提供する項目プロパティと、に対して使用される。例えば、E n t i t y T o G r o u p（「a e b r」）に属する項目のパラメータはI t e m P r o p e r t y（「a e b r」）によって提供され、ここで「a e b r」は自動露光ブラケットセットのためのF o u r C Cである。

10

【0307】

この実施形態の変形例では、2つ以上の撮像モードの組合せに対応するグループ化タイプが定義される。

【0308】

この実施形態の別の変形例では、「c a s e」E n t i t y T o G r o u pがいくつかのc a p t u r e _ m o d eを指定する。capture_modeごとに、及びグループ内の項目ごとに、c a p t u r e _ m o d eと同じ4 C Cを有する項目プロパティが、このcapture_modeの項目特有のパラメータを定義する。

20

【0309】

E n t i t y T o G r o u pとして記述されている場合のグループ化情報の構文は、以下で説明する。

```
aligned(8) class EntityToGnoupBox('case', version, flags) extends FullBox('case', version, flags) {
    unsigned int(32) group_id;
    unsigned int(32) num_entities_in_group;
    for(i=0 i num_entities_in_group i++)
        unsigned int(32) entity_id; //以下のパラメータは、grouping_type'case ' に対する
        共通のパラメータを提供する
    unsigned int(8) num_capture_mode;
    for(i=0 i num_capture_mode i++)
        unsigned int(32) capture_mode; //撮像モードを識別する4 C C
}
```

30

【0310】

この構文と単一のcapture_modeを持つ構文とは、2つの異なるバージョンを使用して単一の定義にグループ化され得ることは知っておくべきである。

【0311】

この変形例は、異なるグループ化情報に対しても使用することができる。

【0312】

40

[V i s u a l S a m p l e E n t r y]

第4の実施形態では、撮像シリーズ内の全ての画像に共通の情報が「p i c t」又は「v i d e」トラックのV i s u a l S a m p l e E n t r y内の新しいボックスに格納される。この新しいボックスは、このサンプルエントリに関連付けられたトラックのサンプルが同じ撮像モード（例えば、ブラケットセット、画像バースト、写真シリーズなど）に属することを示す信号である。サンプル特有の情報（例えば、先の実施形態ではプロパティ情報として説明した）は、専用のメタデータトラックに格納される。例えば、V i s u a l S a m p l e E n t r yは、撮像モードが自動露光ブラケットであることを示す。ブラケットのパラメータ（すなわち、露光構成）は、メタデータトラックのサンプルに記述される。

50

【 0 3 1 3 】

撮像装置は、タイプ「c d s c」のトラック基準を介して、画像サンプルを記述するトラックを、サンプル特有の情報を記述する1つ以上のメタデータトラックに関連付けてもよい。

【 0 3 1 4 】

[タグ付けのための新しい項目プロパティ及びサンプルエントリ]

第5の実施形態では、プロパティ情報が一連の画像に注釈を付けるための情報を記述する。このプロパティ情報は、人間に可読なタグの名前を与えるUTF-8文字におけるnull終端文字列としてラベルパラメータを含んでいる。1つの代替例では、このプロパティが「i p c o」ボックス内の他の項目プロパティとして宣言され、「i p m a」ボックス内の項目に関連付けられることができる。例えば、以下のテーブルでは、ラベルパラメータを包含するItemProperty（又はItemFullProperty）の構文が説明されている。カメラ装置は、このラベルパラメータのコンテンツを、典型的にはユーザに定義されるUTF-8文字の文字列で埋める。1つの代替例では、ラベルが撮影のコンテキストを記述する情報を包含する。例えば、ユーザが文字列を入力してもよく、撮像装置がその撮像の日付及び位置を含む文字列を生成してもよい。例えば、この文字列は、「2013年5月にパリで撮像された、エッフェルタワーの訪問」を包含していてもよい。特に、撮像装置はラベルを自動的に生成するために、その装置上で利用可能な画像認識プログラムを使用してもよい。

【 0 3 1 5 】

```
aligned(8) class Label extends ItemProperty('labl') {
  utf8string label
}
```

【 0 3 1 6 】

1つの代替例では、1つの「t r a k」コンテナに記述された一連の画像に対する1つのSampleEntry又はSampleGroupEntryとしてプロパティ情報が記述される。このコンテキストにおけるプロパティ情報の構文は以下の通りである。このような場合、以下に示すようにこの構文はItemProperty構文に近いものになる。

```
Box type : 'aebr'
Container : VisualSampleEntry
Mandatory: No
Quantity: Zero or one
aligned(8) class Label extends Box('labl') {
  utf8string label
}
```

【 0 3 1 7 】

結果として、ラベルは、トラック内のサンプルのコンテンツを記述するUTF-8文字列になる。

【 0 3 1 8 】

別の実施形態では、特定の文字列を介して（正確な目的なしに）一連の画像が関連させられることをグループ化情報が示す。エンティティグループ化の場合、グループ化タイプは「l a b l」又は「t a g」の4文字コードに等しく設定される。

【 0 3 1 9 】

別の代替例では、タグ又はラベルプロパティを、「l a b l」や「t a g」とは異なるグループ化タイプを有する項目のグループに関連付けることもできる。

【 0 3 2 0 】

別の実施形態では、写真シリーズ又はユーザコレクションプロパティ情報が追加パラメータとしてラベルパラメータを含む。

【 0 3 2 1 】

構文は、例えば次のようになる。

```
aligned(8) class UserDefinedCaptureSeriesProperty extends ItemProperty('ud
cs') {
  utf8string label
}
```

【 0 3 2 2 】

ここで、labelは、ユーザ定義の撮像シリーズの人間に可読な名前を提供するUTF-8文字のnull終端文字列である。

【 0 3 2 3 】

[グループを参照する項目プロパティ関連付け]

第6の実施形態では、プロパティ情報が（例えば、1つのEntityToGroup又はSampleToGroupボックス内のシグナリングを介して）グループ化情報によって1つのグループ内に収集されるいくつかの画像（すなわち、項目又はサンプルのいずれか）に適用される。このような場合、プロパティ情報は、1つの画像に代わり1つのグループに関連付けられる。例えば、ItemPropertyAssociationボックスは、一連の項目をグループ化するグループの識別子（例えば、EntityToGroupのgroup_idパラメータ）を参照することを可能にする。次に、「ipma」ボックスのitem_ID欄は、項目の識別子(item_ID)又はグループ識別子(group_id)のいずれかを参照することができるentity_idによって置き換えられる。例えば、グループ化情報はEntityToGroupである。1つのItemPropertyAssociationボックスが参照できる識別情報は、group_id欄である。この実施形態の利点は、関連付けの繰り返しが回避されるので、各画像のプロパティの記述がよりコンパクトになることである。

【 0 3 2 4 】

以下の表は、ItemPropertyAssociationボックスの新しい構文である。

```
aligned(8) class ItemPropertyAssociation extends FullBox('ipma', version, flags)
{
  unsigned int(32) entry_count;
  for(i = 0; i < entry_count; i++) {
    if (version < 1)
      unsigned int(16) entity_ID;
    else
      unsigned int(32) entity_ID;
    unsigned int(8) association_count;
    for (i=0; i < association_count; i++) {
      bit(1) essential;
      if (flags & 1)
        unsigned int(15) property_index;
      else
        unsigned int(7) property_index;
    }
  }
}
```

【 0 3 2 5 】

ItemPropertyAssociationの異なるパラメータのセマンティクスは、entity_IDに名前変更されるitem_ID欄を除いて、変更されないままである。このentity_IDパラメータは、1つの項目又は1つのEntityToGroupのいずれかを参照してもよい。したがって、entity_IDの値は、1つのgroup_id値又は1つのitem_ID値に等しい必要がある。

10

20

30

40

50

【 0 3 2 6 】

一実施形態では、例えばオーディオサンプル、メタデータ項目など、任意の種類のメディアエンティティのグループをプロパティにリンクするために、同じ機構が使用される。

【 0 3 2 7 】

例えば、撮像装置はユーザコレクション撮像モードを使用し、各画像を1つの項目として記述する。例えば、撮像装置は、4つの画像を格納する。各項目と共有されるプロパティ情報は、前の実施形態のように、撮影者によって設定されたラベルに対応する。H E I Fボックスはこのように4つの項目（ユーザコレクションの各画像に対して1つ）を記述する。

- ・項目 # 1 : ItemId= 1 ;
- ・項目 # 2 : ItemId= 2 ;
- ・項目 # 3 : ItemId= 3 ;
- ・項目 # 4 : ItemId= 4 ;

10

【 0 3 2 8 】

グループ化情報は、4つの項目のセットを収集する「case」グループ化タイプ値を有するEntityToGroupBoxとして記述される。次に、全ての項目に共通するラベルが説明される。この対応するプロパティは、「label」コードによって識別され、例えば、製品基準又は所定の値、例えばそのセット内の第1の画像の撮像日のような、ユーザによって設定されたラベルである。プロパティと項目との関連付けは、1つのItemPropertyAssociationボックスに記述される。この実施形態では、ItemPropertyAssociationボックスが4つの項目に対する単一のエントリのみを包含する。H E I Fライターは、EntityToGroupBoxボックス内で定義された「case」グループで定義されたグループの識別子にプロパティを関連付ける。この相互化は他のプロパティ、例えば、サイズ（「ispe」）又は復号構成（「hvcC」）に適用することができる。このテーブルは、この例のH E I Fファイルの記述の疑似コードである。

20

```
GroupsListBox{
EntityToGroupBox('case') {
group_id='1234'
num_entities_in_group=4
entity_id=1
entity_id=2
entity_id=3
entity_id=4
}
}
ItemPropertyContainerBox {
'ispe'{1920 1080}
'hvcC'{...}
'label' {
label = "Product ref. 32184103"
}
}
ItemPropertyAssociation {
entry_count = 1
entity_id = 1234
association_count = 3
essential = 0, property_index=1 //サイズ
essential = 1, property_index=2 //復号構成
essential = 0, property_index=3 //ラベル
```

30

40

50

}

【 0 3 2 9 】

ライターは、サイズと復号構成を画像ベースで関連付け続け、グループ化タイプに特有のプロパティ（以下の例では「l a b l」）のみを共有してもよい。

ItemPropertyAssociation {

entry_count = 5 //グループに関連付けられたプロパティ

entity_id = 1234

association_count = 1

essential = 0, property_index = 3 //ラベル

//画像項目 # 1 に関連付けられたプロパティ

entity_id = 1

association_count = 2

essential = 0, property_index = 1 //サイズ

essential = 1, property_index = 2 //復号構成

//画像項目 # 2 に関連付けられたプロパティ

//画像項目 # 3 に関連付けられたプロパティ

//画像項目 # 4 に関連付けられたプロパティ

entity_id = 4

association_count = 2

essential = 0, property_index = 1 //サイズ

essential = 1, property_index = 2 //復号構成

【 0 3 3 0 】

上の例から分かるように、画像又はエンティティのグループへの1つ以上のプロパティの関連付けを可能にする、画像プロパティ関連付けマップを有することは、画像ファイルの冗長な記述の減少をもたらす。

【 0 3 3 1 】

[項目リファレンス]

第7の実施形態では、撮像された一連の画像間の関係がItemReferenceとして1つの撮像シリーズで記述される。ItemReferenceBoxはHEIFファイルの2つの項目間の基準を記述し、reference_typeパラメータを介して各関連付けに基準のタイプを関連付ける。この実施形態では、各撮像モードに対して1つの基準タイプが定義される。所与の撮像モードに対して使用するreference_typeの4文字コードは、grouping_typeに対して記述されているものと同じである。

【 0 3 3 2 】

その原理は、その撮像順序でその装置によって撮像された各画像間に1つの項目基準を設定することである。したがって、例えば、一連の画像のうちの第2の画像は、一連の画像のうちの第1の画像への基準を記述し、第3の画像は第2の画像への基準を記述し、以下同様である。

【 0 3 3 3 】

reference_type値に応じて、HEIFパーサは1番目と2番目の項目の関係を決定する。

【 0 3 3 4 】

例えば、reference_typeが、

- ・「p a n o」に等しい場合、第2の項目は、パノラマの撮像パターン内の第1の項目に続く画像である。この撮像パターンが左から右である場合、第2の項目は第1の項目の右にある。デフォルトでは、左から右へのパターンが想定される。

- ・「p a n h」は「p a n o」の代替である場合、第2の項目は、パノラマの撮像パターン内の第1の項目に水平に続く画像である。この撮像パターンが左から右である場合、第2の項目は第1の項目の右にある。デフォルトでは、左から右へのパターンが想定される。別の実施形態では、「p a n H」reference_typeが右から左へのパノラマに使用される。

10

20

30

40

50

・「p a n v」に等しい場合、第2の項目は、パノラマの撮像パターン内の第1の項目に水平に続く画像である。この撮像パターンが上から下である場合、第2の項目は第1の項目の下にある。デフォルトでは、上から下へのパターンが想定される。別の実施形態では、「p a n V」reference_typeが下から上へのパノラマに使用される。-「p a n h」、「p a n H」、及び「p a n v」、「p a n V」を、格子パノラマを記述するために組み合わせることができる。各行は、「p a n h」又は「p a n H」reference_typeを使用して記述される。2つの連続する行間の関係は、2つの行の最初の画像間の「p a n v」又は「p a n V」reference_typeを使用して記述される。

・これらのreference_typeは、「r i g h」、「l e f t」、「t o p」、「b o t t」と名付けることもできる。

・これらのreference_typeは、立方体パノラマ又はより複雑なパノラマを記述するために使用してもよい。

・「i m b u」又は「p s e r」又は「t i l a」又は「b r s t」に等しい場合、第2の項目は、画像のセットの撮像の順序内の第1の項目に続く画像である。

・「t s y n」に等しい場合、第2の項目は第1の項目と同じ時間に撮像された。

・「b r a k」に等しい場合、第2の項目は、第1の項目に続くブラケットセット内の画像である。

・「a e b r」に等しい場合、第2の項目は、第1の項目に続く露光ブラケットセット内の画像である。その結果、露光絞りの昇順でブラケットを行った場合の第1の項目よりも長い露光時間としての第2の項目。

・「w b b r」に等しい場合、第2の項目は、第1の項目に続くホワイトバランスブラケットセット内の画像である。その結果、色温度の昇順でブラケットを行った場合の第1の項目よりも高い色温度としての第2の項目。

・「f o b r」に等しい場合、第2の項目は、第1の項目に続くフォーカスブラケットセット内の画像である。その結果、フォーカス距離の昇順でブラケットを行った場合の第1の項目よりも遠いフォーカス距離としての第2の項目。

・「m u l t」に等しい場合、第2の項目は、同じ多重露光セットの別の項目である。

・「h d r」に等しい場合、第2の項目は、同じH D R項目セットの別の項目である。

・「n o i s」に等しい場合、第2の項目は、同じノイズ低減項目セットの別の項目である。

・「d a r k」に等しい場合、第2の項目は、第1の画像からセンサ関連ノイズを除去するために使用することができる暗い画像である。

・「f l a t」に等しい場合、第2の項目は、第1の画像の口径食を補償するために使用することができるフラットな画像である。

【0335】

この原理は、他のブラケットタイプについても同様である。

【0336】

別の実施形態では、撮像シリーズからの画像のセットは、新しい画像を生成するために撮像装置によってさらに処理される。例えば、撮像装置は、自動露光ブラケットで撮像された一連の画像からH D R画像を生成してもよい。そのような場合、項目基準は、自動露光ブラケットにおいて、H D R項目と一連の画像のグループの識別子との間でシグナリングされる。他の実施例は、ノイズ低減、長秒時露光ノイズ低減、口径食補正、超解像、多重露光画像などがある。その結果、I t e m R e f e r e n c e B o x構文は次のようになる。

```
aligned(8) class SingleItemTypeReferenceBox(referenceType) extends Box(referenceType) {
    unsigned int(16) from_item_ID;
    unsigned int(16) reference_count;
    for (j=0 j reference_count j++) {
        unsigned int(16) to_entity_ID;
```

```

}
}
aligned(8) class SingleItemTypeReferenceBoxLarge(referenceType) extends Bo
x(referenceType) {
unsigned int(32) from_item_ID;
unsigned int(16) reference_count;
for (j=0 j reference_count j++) {
unsigned int(32) to_entity_ID;
}
}
aligned(8) class ItemReferenceBox extends FullBox('iref', version, 0) {
if (version==0) {
SingleItemTypeReferenceBox references [];
} else if (version==1) {
SingleItemTypeReferenceBoxLarge references [];
}
}

```

10

【 0 3 3 7 】

新しいパラメータto_entity_IDは、項目又はグループの識別子である。この利点は、H E I F パーサが、画像のグループから1つの項目が計算されることを決定できることである。

20

【 0 3 3 8 】

処理された画像を画像のグループに関連づけるために、いくつかのタイプの基準を定義することができる。これらのreference_typesは、項目とグループの間、又は項目と他のいくつかの項目の間で使用できる。

【 0 3 3 9 】

これらの新しいタイプは、例えば以下のである。

- ・「P a n o」：この項目は、参照された項目から生成されたパノラマ項目である。

- ・「S r e s」：この項目は、参照された項目から生成された超解像項目である。

- ・「M u l t」：この項目は、参照された項目から生成された多重露光項目である。

- ・「H d r」：この項目は、参照された項目から生成されたH D R項目である。

30

- ・「N o i s」：この項目は、参照された項目から生成されるノイズが低減された項目である。

- ・「D a r k」：この項目は、参照された項目を使用して長秒時露光関連ノイズが除去された項目である。第1の参照項目は原稿の長秒時露光画像であり、第2の参照項目は暗画像である。

- ・「F l a t」：この項目は、参照された項目を使用して口径食が補償された項目である。第1の参照項目はオリジナル画像であり、一方第2の参照項目はフラットな画像である。

【 0 3 4 0 】

[撮像モードごとに1つのグループ化タイプ及び1つのプロパティ情報]

一実施形態では、第1の実施形態におけるように、グループ化タイプは撮像モードそれぞれに対して定義される。さらに、第2の実施形態のように、プロパティ情報は撮像モードそれぞれに対して定義される。このように、撮像モードのパラメータは、対応するプロパティ情報内に記述される。

40

【 0 3 4 1 】

すなわち、グループ化タイプ及びプロパティ情報を以下のように定義することができる。

【 0 3 4 2 】

「b r s t」エンティティグループは、画像項目の設定が時間的に増加する順序でバーストを形成することを示すことを可能にする。

【 0 3 4 3 】

我々は、2つの連続する画像項目間の撮像時間差を文書化するために使用することのでき

50

る、関連付けられた「b r s t」記述項目プロパティを導入することによって、このグループを構築することを提案する。この項目プロパティは例えば、タイムラプスをより完全に記述するために有用である。

【 0 3 4 4 】

[画像バースト情報]

[定義]

画像バースト「b r s t」記述項目プロパティは、画像項目のバースト内の関連付けられた画像項目と前の画像項目との間の撮像時間差を定義する。関連付けられた画像項目を参照するエンティティグループ「b r s t」は1つ存在し、1つのみ存在する必要がある。

【 0 3 4 5 】

[構文]

```
aligned (8) class ImageBurstProperty extends ItemProperty('brst') {
  unsigned int (32) time_delta;
}
```

【 0 3 4 6 】

[セマンティクス]

time_deltaは、画像項目とバースト内の以前の項目との間に経過した100分の1秒の数である。この値は、関連付けられた画像項目がタイプ「b r s t」のエンティティグループ内の最初の画像項目である場合、値0として解釈されることになる。

【 0 3 4 7 】

[画像バースト情報代替例]

[定義]

画像バースト「b r s t」記述項目プロパティは、画像項目のバースト内の関連付けられた画像項目の撮像時間を定義する。画像項目のバースト内の最初の画像項目は、画像項目のバーストが開始し、そこから撮像時間が測定される時間ゼロを表す。関連付けられた画像項目を参照するエンティティグループ「b r s t」は1つ存在し、1つのみ存在する必要がある。

【 0 3 4 8 】

[構文]

```
aligned (8) class ImageBurstProperty extends ItemProperty('brst') {
  unsigned int (32) capture_time;
}
```

【 0 3 4 9 】

[セマンティクス]

capture_timeは、画像項目とバースト内の最初の画像項目との間に経過した100分の1秒の数である。この値は、関連付けられた画像項目がタイプ「b r s t」の関連付けられたエンティティグループ内の最初の画像項目である場合、値0として解釈されることになる。

【 0 3 5 0 】

[ブラケットセット / 論理グループ]

[動機]

ブラケットは、異なるカメラ設定で同じコンテンツを撮像することを可能にする。最も一般的なケースは、シーンの全ての部分が、画像の少なくとも1つにおいて露光ブラケットを使用して正しい露光で撮影されるであろうことを確かにすることである。通常、結果として得られる画像のセットは（例えば、HDR画像を作成することによって）最終結果を得るために処理される。

【 0 3 5 1 】

画像間のブラケット関係を明示的に維持し、明示的にすることは、全ての写真処理ワークフローの間、撮影者の最初の意図を保持するために非常に有用である。さらに、異なる画像に関連付けられたメタデータから画像項目を決定することが困難であり得る（利用可能

10

20

30

40

50

な E X I F 欄に基づいていくつかの画像項目から複数の値を計算し交差させること、又は各カメラメーカに固有のメーカノートを使用することなどが必要であり得る)ので、ブラケットセット内の画像項目を識別する汎用パラメータ値を指定するプロパティを提供することは、クライアントによる画像項目の選択を容易にするためにも有用である。

【0352】

ブラケットに関連して、パノラマ作成を意図して複数の画像を撮像することができる。ブラケットに関しては、画像間のこのパノラマ関連のリンクを明示的に維持し、明示的にすることが非常に有用である。また、パノラマの特性を指定するプロパティを提供することは、画像をパノラマに処理するのに役立つ。

【0353】

以下のセクションは、T u C に記述されているエンティティグループ及びサンプルグループのための規範的なテキストを提案する。対応する項目プロパティのための規範的なテキストは後で提案される。

【0354】

[自動露光ブラケット]

[「a e b r」エンティティグループ]

自動露光ブラケットエンティティグループ(「a e b r」)は、異なる露光設定で撮像された画像項目のセットを示す。エンティティグループ内の各画像項目に対する相対露光設定は、自動露光ブラケット項目プロパティ「a e b r」を使用して定義されるべきである。

【0355】

[「a e b r」サンプルグループ]

[定義]

自動露光ブラケットサンプルグループ(「a e b r」)は、異なる露光設定で撮像されたサンプルのセットを示す。「a e b r」タイプのサンプルグループがいくつかあってもよく、そのそれぞれは異なるgrouping_type_parameterの値を有する。A u t o E x p o s u r e B r a c k e t i n g E n t r y は、サンプルグループ内のこのエントリに関連付けられたサンプルに対する相対露光設定を提供する。

【0356】

[構文]

```
aligned (8) class AutoExposureBracketingEntry extends VisualSampleGroupEntry('aebr') {
int (8) exposure_step;
int (8) exposure_numerator;
}
```

【0357】

[セマンティクス]

exposure_stepは、露光ブラケットの間使用される増分ステップを指定する整数値である。1に等しい場合は絞りインクリメントがそのまま使用され、2に等しい場合は1/2絞りインクリメントが使用され、3に等しい場合は1/3絞りインクリメントが使用され、4に等しい場合は1/4絞りインクリメントが使用される。

【0358】

exposure_numeratorは、項目の露光値絞りを計算するために使用される露光分子を指定する整数値である。

【0359】

デフォルトの自動露光カメラ設定と比較した画像項目の露光値変動は、exposure_numerator/exposure_stepとして計算される絞りの数として表される。

【0360】

[ホワイトバランスブラケット]

[「w b b r」エンティティグループ]

ホワイトバランスブラケットエンティティグループ(「w b b r」)は、異なるホワイト

10

20

30

40

50

バランス設定で撮像された画像項目のセットを示す。エンティティグループ内の各画像項目に対する相対ホワイトバランス設定は、ホワイトバランスブラケット項目プロパティ「w b b r」を使用して定義されるべきである。

【 0 3 6 1 】

[「 w b b r 」 サンプルグループ]

[定義]

ホワイトバランスブラケットサンプルグループ (「 w b b r 」) は、異なるホワイトバランス設定で撮像されたサンプルのセットを示す。「w b b r」タイプのサンプルグループがいくつかあってもよく、そのそれぞれは異なるgrouping_type_parameterの値を有する。WhiteBalanceBracketingEntryは、サンプルグループ内のこのエントリに関連付けられたサンプルについて、青 / 黄色バイアス及び / 又はマゼンタ / 緑バイアスに関するホワイトバランス補償を提供する。

10

【 0 3 6 2 】

[構文]

```
aligned(8) class WhiteBalanceBracketingEntry extends VisualSampleGroupEntry('wbbr') {
int(8) blue_amber;
int(8) green_magenta;
}
```

【 0 3 6 3 】

20

[セマンティクス]

blue_amberは、青 - 黄色軸のホワイトバランス補償の値である。

【 0 3 6 4 】

green_magentaは、緑 - マゼンタ軸のホワイトバランス補償の値である。

【 0 3 6 5 】

[フォーカスブラケット]

[「 f o b r 」 エンティティグループ]

フォーカスブラケット露光ブラケットエンティティグループ (「 f o b r 」) は、異なるフォーカス設定で撮像された画像項目のセットを示す。エンティティグループ内の各画像項目の相対フォーカス設定は、フォーカスブラケット項目プロパティ「f o b r」を使用して定義されるべきである。

30

【 0 3 6 6 】

[「 f o b r 」 サンプルグループ]

[定義]

フォーカスブラケット露光ブラケットサンプルグループ (「 f o b r 」) は、異なるフォーカス設定で撮像されたサンプルのセットを示す。「f o b r」タイプのサンプルグループがいくつかあってもよく、そのそれぞれは異なるgrouping_type_parameterの値を有する。FocusBracketingEntryは、サンプルグループ内のこのエントリに関連付けられたサンプルに対する相対フォーカス設定を提供する。

【 0 3 6 7 】

40

[構文]

```
aligned(8) class FocusBracketingEntry extends VisualSampleGroupEntry('fobr') {
unsigned int(32) focus_distance_numerator;
unsigned int(32) focus_distance_denominator;
}
```

【 0 3 6 8 】

[セマンティクス]

フォーカス距離は、focus_distanse_numeratorとfocus_distanse_denominatorの比としてメートル単位で表される。

【 0 3 6 9 】

50

[フラッシュ露光ブラケット撮影]

[「 a f b r 」 エンティティグループ]

フラッシュ露光ブラケットエンティティグループ (「 a f b r 」) は、異なるフラッシュ露光設定で撮像された画像項目のセットを示す。エンティティグループ内の各画像項目に対する相対フラッシュ露光設定は、フラッシュ露光ブラケット項目プロパティ「 a f b r 」を使用して定義されるべきである。

【 0 3 7 0 】

[「 a f b r 」 サンプルグループ]

[定義]

フラッシュ露光ブラケットサンプルグループ (「 a f b r 」) は、異なるフラッシュ露光設定で撮像されたサンプルのセットを示す。「 a f b r 」型のサンプルグループがいくつかあってもよく、そのそれぞれは異なるgrouping_type_parameterの値を有する。FlashExposureBracketingEntryは、サンプルグループ内のこのエントリに関連付けられたサンプルの相対フラッシュ露光設定を提供する。

10

【 0 3 7 1 】

[構文]

```
aligned(8) class FlashExposureBracketingEntry extends VisualSampleGroupEntry('afbr') {
int(8) flash_exposure_numerator;
int(8) flash_exposure_denominator;
}
```

20

【 0 3 7 2 】

[セマンティクス]

カメラ設定と比較したサンプルのフラッシュ露光変動は、絞りの数として表され、flash_exposure_numerator / flash_exposure_stepとして計算される。

【 0 3 7 3 】

[被写界深度ブラケット]

[「 d o b r 」 エンティティグループ]

被写界深度ブラケットエンティティグループ (「 d o b r 」) は、異なる被写界深度設定で撮像された画像項目のセットを示す。エンティティグループ内の各画像項目の相対被写界深度設定は、被写界深度ブラケット項目プロパティ「 d o b r 」を使用して定義されるべきである。

30

【 0 3 7 4 】

[「 d o b r 」 サンプルグループ]

[定義]

被写界深度ブラケットサンプルグループ (「 d o b r 」) は、異なる被写界深度設定で撮像されたサンプルのセットを示す。「 d o b r 」タイプのサンプルグループがいくつかあってもよく、そのそれぞれは異なるgrouping_type_parameterの値を有する。DepthOfFieldBracketingEntryは、サンプルグループ内のこのエントリに関連付けられたサンプルに対する相対被写界深度設定を提供する。

40

【 0 3 7 5 】

[構文]

```
aligned(8) class DepthOfFieldBracketingEntry extends VisualSampleGroupEntry('dobr') {
int(8) f_stop_numerator;
int(8) f_stop_denominator;
}
```

【 0 3 7 6 】

[セマンティクス]

被写界深度変動は、絞り数の口径変化として表され、f_stop_numerator / f_stop_deno

50

minatorとして計算される。

【 0 3 7 7 】

[パノラマ]

[「 p a n o 」 エンティティグループ]

パノラマエンティティグループ (「 p a n o 」) は、パノラマを作成するために撮像された画像項目のセットを示す。パノラマ内のエンティティグループからの各画像項目の位置は、パノラマ項目プロパティ 「 p a n o 」 を使用して定義されるべきである。

【 0 3 7 8 】

[「 p a n o 」 サンプルグループ]

[定義]

パノラマサンプルグループ (「 p a n o 」) は、パノラマを作成するために撮像されたサンプルのセットを示す。「 a e b r 」タイプのサンプルグループがいくつかあってもよく、それぞれは異なるgrouping_type_parameterの値を有する。P a n o r a m a E n t r y は、サンプルグループ内のこのエントリに関連付けられたサンプルの位置を提供する。

【 0 3 7 9 】

[構文]

```
aligned(8) class PanoramaEntry extends VisualSampleGroupEntry('pano') {
  unsigned int(1) panorama_direction unsigned int(15) frame_number
}
```

【 0 3 8 0 】

[セマンティクス]

panorama_directionはパノラマ方向をシグナリングする 1 ビットのフラグ (0 : 左から右、 1 : 右から左) であり、frame_numberはパノラマ方向に従うフレーム番号を (低い方から高い方へと) 表す整数である。

【 0 3 8 1 】

[項目プロパティ]

[動機]

以下のセクションは、ブラケット、パノラマ、及びバーストに関連するエンティティグループを指定するために、項目プロパティの規範的なテキストを提案する。

【 0 3 8 2 】

[自動露光ブラケット情報]

[定義]

自動露光ブラケット記述項目プロパティは、カメラ設定に対して相対的に関連付けられた画像項目の露光変動を定義する。

【 0 3 8 3 】

これは、「 a e b r 」エンティティグループに含まれる画像項目のプロパティを指定するために使用される。

【 0 3 8 4 】

[構文]

```
aligned (8) class AutoExposureBracketingProperty extends ItemProperty('aeb
r') {
  int (8) exposure_step;
  int (8) exposure_numerator;
}
```

【 0 3 8 5 】

[セマンティクス]

exposure_stepは、露光ブラケットの間使用される増分ステップを指定する整数値である。1 に等しい場合は絞りインクリメントがそのまま使用され、2 に等しい場合は 1 / 2 絞りインクリメントが使用され、3 に等しい場合は 1 / 3 絞りインクリメントが使用され、4 に等しい場合は 1 / 4 絞りインクリメントが使用される。

10

20

30

40

50

【 0 3 8 6 】

exposure_numeratorは、項目の露光値絞りを計算するために使用される露光分子を指定する整数値である。

【 0 3 8 7 】

カメラ設定と比較した関連付けられた画像項目の露光値変動は、絞りの数として表され、exposure_numerator / exposure_stepとして計算される。

【 0 3 8 8 】

[ホワイトバランスブラケット情報]

[定義]

ホワイトバランスブラケット記述項目プロパティは、カメラ設定に対して相対的に関連付けられた画像項目に適用される青 / 黄色バイアス及び / 又はマゼンタ / 緑バイアスに関するホワイトバランス補償を定義する。

10

【 0 3 8 9 】

これは、「w b b r」エンティティグループに含まれる画像項目のプロパティを指定するために使用される。

【 0 3 9 0 】

[構文]

```
aligned (8) class WhiteBalanceBracketingProperty extends ItemProperty ('wb
br') {
int (8) blue_amber;
int (8) green_magenta;
}
```

20

【 0 3 9 1 】

[セマンティクス]

blue_amberは、青 - 黄軸におけるホワイトバランス補償の値である。

【 0 3 9 2 】

green_magentaは、緑 - マゼンタ軸におけるホワイトバランス補償の値である。

【 0 3 9 3 】

[フォーカスブラケット情報]

[定義]

フォーカスブラケット記述項目プロパティは、カメラ設定に対して相対的に関連付けられた画像項目のフォーカス変動を定義する。

30

【 0 3 9 4 】

これは、「f o b r」エンティティグループに含まれる画像項目のプロパティを指定するために使用される。

【 0 3 9 5 】

[構文]

```
aligned (8) class FocusBracketingProperty extends ItemProperty('fobr') {
unsigned int(32) focus_distance_numerator;
unsigned int(32) focus_distance_denominator;
}
```

40

【 0 3 9 6 】

[セマンティクス]

フォーカス距離は、focus_distanse_numeratorとfocus_distanse_denominatorの比としてメートル単位で表される。

【 0 3 9 7 】

[フラッシュ露光ブラケット情報]

[定義]

フラッシュ露光ブラケット記述項目プロパティは、カメラ設定に対して相対的に関連付けられた画像項目のフラッシュ露光変動を定義する。

50

【 0 3 9 8 】

これは、「a f b r」エンティティグループに含まれる画像項目のプロパティを指定するために使用される。

【 0 3 9 9 】

[構文]

```
aligned (8) class FlashExposureBracketingProperty extends ItemProperty('afb
r') {
int (8) flash_exposure_numerator;
int (8) flash_exposure_denominator;
}
```

10

【 0 4 0 0 】

[セマンティクス]

カメラ設定に対する相対的な画像項目のフラッシュ露光変動は、絞りの数として表され、 $\text{flash_exposure_numerator} / \text{flash_exposure_step}$ として計算される。

【 0 4 0 1 】

[被写界深度ブラケット情報]

[定義]

被写界深度ブラケット記述項目プロパティは、カメラ設定に対して相対的に関連付けられた画像項目の被写界深度変動を定義する。これは、口径変化として表現される。これは、「d o b r」エンティティグループに含まれる画像項目のプロパティを指定するために使用される。

20

【 0 4 0 2 】

[構文]

```
aligned (8) class DepthOfFieldBracketingProperty extends ItemProperty('dobr') {
int (8) f_stop_numerator;
int (8) f_stop_denominator;
}
```

【 0 4 0 3 】

[セマンティクス]

被写界深度変動は、絞り数の口径変化として表され、 $\text{f_stop_numerator} / \text{f_stop_denominator}$ として計算される。

30

【 0 4 0 4 】

[パノラマ情報]

[定義]

パノラマ記述項目プロパティは、パノラマ内の関連付けられた画像項目の位置を定義する。これは、「p a n o」エンティティグループに含まれる画像項目のプロパティを指定するために使用される。関連付けられた画像項目を参照するエンティティグループ「p a n o」は1つ存在し、1つのみ存在する必要がある。

【 0 4 0 5 】

[構文]

```
aligned (8) class PanoramaProperty extends ItemProperty('pano') {
unsigned int(1) panorama_direction;
unsigned int(15) frame_number;
}
```

40

【 0 4 0 6 】

[セマンティクス]

`panorama_direction`はパノラマ方向をシグナリングする1ビットのフラグ（0：左から右、1：右から左）であり、`frame_number`はパノラマ方向に従うフレーム番号を（低い方から高い方へと）表す整数である。

【 0 4 0 7 】

50

図3は、図2のカプセル化処理によって生成されるHEIFファイル解析処理の主なステップを示す。デコード処理は、一連の画像を有するHEIFファイルの解析によって開始する。ステップ301で、グループ化情報が解析される。一連の画像の撮像モードは、グループ化情報が存在するとき、すなわち、HEIFファイルが先に述べた値の1つに等しいgrouping_typeを有するEntityToGroup又はTrackGroup又はSampleToGroupボックスを含むとき、ステップ303で決定される。第1の代替例では、grouping_typeパラメータは撮像モードを直接指定する。第2の代替例では、グループ化情報は、画像のセットが撮像シリーズグループに所属する（グループ化タイプが「case」に等しい）ことをシグナリングする。この場合、プロパティ情報はステップ302において、ItemProperty、ItemFullProperty、又はVisualSampleEntry若しくはVisualSampleGroupEntryのいずれかのボックスで解析され、ステップ303において、画像のグループの撮像モード、並びにそのセット及び撮像シリーズの各画像に関連関連付けられたパラメータを決定する。

【0408】

撮像モードがブラケット撮像モードに対応する場合、デコーダはステップ304において、HEIFファイルが一連のブラケット画像を包含することをプレーヤに通知する。そのような場合、このアプリケーションは、異なるブラケット代替例を眺めることを可能にするGUIインターフェースを提供する。一実施形態では、このインターフェースは、各画像に関連付けられた撮像の特性を抽出するために、ItemPropertiesなどのプロパティ情報内に提供される情報を提供する。特に、自動露光ブラケット撮影では、ユーザが適切なショットを選択できるように、各ショットの露光絞りがステップ305で表示される。好ましい露光が選択されると、復号装置は、選択された画像を「一次項目」としてマークするようにHEIFファイルを修正してもよい。

【0409】

撮像モードがパノラマ画像に対応する場合、デコーダはステップ304において、ユーザがナビゲートできる一連の画像をHEIFファイルが包含することをプレーヤに通知する。このGUIインターフェースは、ユーザが空間的構成として画像間をナビゲートできるように、特定のGUI要素を示してもよい。プレーヤは、パターンに適合されたナビゲーションインターフェースを生成するように、（例えば、左から右への）セット画像の撮像のパターンを抽出するために、プロパティ情報を解析する。例えば、撮像のパターンが左から右である場合、GUIインターフェースは、HEIFファイルの項目間をナビゲーションすることを可能にする水平ナビゲーション矢印を提供する。

【0410】

撮像モードが写真シリーズ又は画像バーストに対応する場合、デコーダはステップ304において、例えば、写真シリーズ又は画像バーストグループの全ての画像間のダイアログを開始するようにプレーヤに通知する。一実施形態では、ダイアログ内の各画像の表示時間は、画像バーストグループのプロパティ情報内に指定される時間隔の関数である。

【0411】

一実施形態では、プレーヤは、ユーザがフォトシリーズのコンテンツを迅速に識別できるように、ダイアログの始めに、又は各画像内の透かしとして、プロパティ情報内に提供されるラベル情報を表示する。別の実施形態では、ユーザは、フォトシリーズグループの一連の画像から、好ましい画像として1つの画像を選択してもよい。このような場合、この好ましい画像は一次項目としてマークされる。別の実施形態では、ユーザは、フォトシリーズから好ましい画像としていくつかの画像を選択してもよい。このような場合、プレーヤは選択された画像を有する新しいフォトシリーズグループを作成し、同じラベルプロパティ情報を関連付ける。プロパティ情報内の新しいブールパラメータは、その写真シリーズが写真シリーズの一次グループであることを示す。

【0412】

一実施形態によれば、EntityToGroupBoxから継承された新しいEnti

10

20

30

40

50

`EntityToGroup`は、全ての撮像又はコレクションモードをカバーする固有の汎用の`grouping_type`値を用いて定義される。特定のタイプの撮像又はコレクションモードは、`EntityToGroupBox`の追加属性として定義される。

【0413】

例えば、汎用の「`brak`」（ブラケット用）又は「`case`」（撮像シリーズ用）又は「`lgrp`」（論理グループ化用）グループ化タイプ（又は、すでに使用されていない任意の4文字コードが同等のセマンティックで使用されてもよい）を有する新しい`EntityToGroup`は、特定の撮像又はコレクションモードに従って複数のエンティティをグループ化するために定義されてもよい。すなわち例えば、自動露光ブラケット、ホワイトバランスブラケット、フォーカスブラケット、フラッシュ露光ブラケット、被写界深度ブラケット、ISOブラケット、お気に入りコレクション、アルバムコレクション、又はユーザ定義の撮像シリーズなどの、特定のタイプの撮像又はコレクションモードは、`EntityToGroupBox`（「`case`」）又は同等の`grouping_type`の新しいパラメータ`capture_mode`（又は`collection_mode`、若しくは`grouping_mode`若しくは`logical_grouping_type`）を使用してシグナリングされてもよい。

10

【0414】

`EntityToGroup`として記述されている場合のグループ化情報の構文の例が以下に記述される。

```
aligned(8) class EntityToGroupBox('case', version, flags) { extends FullBox('case', version, flags) {
    unsigned int(32) group_id;
    unsigned int(32) num_entities_in_group;
    for(i=0; i < num_entities_in_group; i++)
        unsigned int(32) entity_id; //以下のパラメータはgrouping_type「case」に対する共通のパラメータである
    unsigned int(32) capture_mode; //撮像又はコレクションモードを識別する4CC
}
```

20

【0415】

ここで、`capture_mode`は撮像又はコレクションモードを識別する。

【0416】

`EntityToGroupBox`からの明示的な継承と同等の構文は、以下で記述される。

30

```
aligned(8) class CaptureSeriesEntityToGroupBox extends EntityToGroupBox('case', version, flags) {
    unsigned int(32) capture_mode;
}
```

【0417】

ある実施形態では、`CaptureSeriesEntityToGroupBox`は`LogicalEntityToGroupBox`又は`BracketingEntityToGroupBox`と名付けられてもよい。

40

【0418】

あるいは、撮像又は収集モードごとに新しい4CCを定義するのではなく、`capture_mode`パラメータを以下の表のインデックスとして定義してもよい。

【0419】

撮像又は収集モードを識別するために、次の4CCコードを定義できる。

【表 2】

撮像モードタイプ	4ccコード
自動露光ブラケット	「aebr」
ホワイトバランスブラケット	「wbbr」
フォーカスブラケット	「fobr」
フラッシュ露光ブラケット	「afbr」
被写界深度ブラケット	「dobr」
パノラマ	「pano」
ユーザ定義の撮像シリーズ	「udcs」
タイムラプス	「tila」
超解像	「sres」
多重露光	「mult」
ノイズ低減	「nois」
長秒時露光ノイズ低減	「dark」
口径食補償	「flat」
HDR	「hdr」
アルバムコレクション	「albc」
お気に入りコレクション	「favc」

10

20

【0420】

30

あるいは、変形例では、特定のgrouping_typeを有する新しいEntityToGroupBoxを、自動露光ブラケットのための「aebr」、ホワイトバランスブラケットのための「wbbr」など、上記の表に列挙された特定の撮像又はコレクションモードごとに定義してもよい。

【0421】

新しいEntityToGroupBoxの構文の例が以下に記述される（撮像モード又はコレクションモードごとに同様の構文が導出されてもよい）。

```
aligned(8) class AlbumCollectionEntityToGroupBox extends EntityToGroupBox
('albc', version, flags) {
```

```
}
```

40

```
aligned(8) class FavoriteCollectionEntityToGroupBox extends EntityToGroupB
ox('favc', version, flags) {
```

```
}
```

```
aligned(8) class AutoExposureBracketingEntityToGroupBox extends EntityToG
roupBox('aebr', version, flags) {
```

```
}
```

【0422】

この第1の実施形態の代替例によれば、上記の新しいEntityToGroupBoxesに基づくグループ化情報は、人間に可読な名前、タグ、若しくはエンティティのグループのコンテンツの記述を与えるUTF-8文字でnull終端文字列を提供する1つ以上の

50

追加のタグ又はラベルパラメータを含んでもよい。それらはまた、位置情報パラメータ（例えば、GPS座標又はその位置の人間に可読な記述）と、他のヌル終端文字列パラメータに包含されるテキストの言語を表す言語情報パラメータとを随意に含んでもよい。

【0423】

それらはまた、グループ化情報の一意の識別子（group_uuid又はlogical_group_idなど）を提供するパラメータを随意に含んでもよい。この一意の識別子は、HEIFファイル内部で、又は複数のHEIFファイルに渡って、複数のエンティティのグループ（例えば、同じアルバムに属するディレクトリ内の複数の画像ファイル）を相互に関連付けるために使用されてもよい。この識別子の一意性を確保するために、特に異なる装置、ユーザ、又はベンダから識別子が割り当てられている場合、この識別子はRFC 4122で指定されているUniversally Unique Identifier(UUID)を使用して定義される。

10

【0424】

EntityToGroupとして記述されている場合の汎用グループ化情報の構文の例が以下に記述される（全てのオプションパラメータを含む）。

```
aligned(8) class CaptureSeriesEntityToGroupBox extends EntityToGroupBox('case', version = 0, flags = 0){
    unsigned int(32) capture_mode;
    unsigned int(8)[16] group_uuid;
    utf8string group_name;
    utf8string group_description;
    utf8string group_tags;
    utf8string lang;
}
```

20

【0425】

ここで、

group_nameは、エンティティのグループの人間に可読な名前を包含するnull終端UTF-8文字の文字列である。

group_descriptionは、エンティティのグループの人間に可読な記述を包含するnull終端UTF-8文字の文字列である。

group_tagsは、エンティティのグループに関連するカンマ区切りのタグを包含するnull終端UTF-8文字の文字列である。

30

langは、「en-US」、「fr-FR」、「zh-CN」などのRFC 5646準拠の言語タグ文字列を包含する文字の文字列で、group_name、group_description、及びgroup_tagsに包含されるテキストの言語を表す。langが空の場合、言語はunknown/undefinedである。

【0426】

汎用のgrouping_typeの代わりに特定のgrouping_typeが使用される場合、特定のEntityToGroupBoxesの構文の例は次のようになる（上記と同じパラメータのセマンティクスを使用）。

```
aligned(8) class AlbumCollectionEntityToGroupBox extends EntityToGroupBox('albc', version, flags) {
    unsigned int(8)[16] group_uuid;
    utf8string group_name;
    utf8string group_description;
    utf8string group_tag;
    utf8string lang;
}

aligned(8) class FavouriteCollectionEntityToGroupBox extends EntityToGroupBox('favc', version, flags) {
    unsigned int(8)[16] group_uuid;
```

40

50

```

utf8string group_name;
utf8string group_description;
utf8string group_tag;
utf8string lang;
}
aligned(8) class AutoExposureBracketingEntityToGroupBox extends EntityToG
roupBox('aebr', version, flags) {
unsigned int(8)[16] group_uuid;
utf8string group_name;
utf8string group_description;
utf8string group_tag;
utf8string lang;
}

```

10

【 0 4 2 7 】

撮像又はコレクションモードごとに、同様の構文が導出されてもよい。

【 0 4 2 8 】

上記の構文は、人間に可読な文字列パラメータに対して1つの言語のみを表現できる。ユーザが表示可能なテキストの代替的な国際化をサポートするために、そのような人間に可読な文字列を複数の言語（例えば、日本語及びフランス語の両方）で提供することが望ましい場合がある。

20

【 0 4 2 9 】

変形例では、新しい `EntityToGroupBoxes` は、それに関連付けられた言語での、代替の人間に可読な文字列パラメータのリストを包含する。

【 0 4 3 0 】

`EntityToGroup` として記述される場合の汎用グループ化情報の構文は、以下に記述される（全てのオプションパラメータを含む）。

```

aligned(8) class CaptureSeriesEntityToGroupBox extends EntityToGroupBox('c
ase', version = 0, flags = 0){
unsigned int(32) capture_mode;
unsigned int(8)[16] group_uuid;
unsigned int(16) entry_count;
for (i=0; i < entry_count; i++){
utf8string group_name;
utf8string group_description;
utf8string group_tags ;
utf8string lang;
}
}

```

30

【 0 4 3 1 】

ここで、`entry_count` は、ユーザが表示可能な代替テキストの数を提供する。

40

【 0 4 3 2 】

同様に、特定の `grouping_type` を有する `EntityToGroupBoxes` の構文は、次のようであってもいい（上記と同じパラメータのセマンティクスを有する）。

```

aligned(8) class AlbumCollectionEntityToGroupBox extends EntityToGroupBox
( 'albc', version = 0, flags = 0){
unsigned int(8)[16] group_uuid;
unsigned int(16) entry_count;
for (i=0; i < entry_count; i++){
utf8string group_name;
utf8string group_description;

```

50

```

utf8string group_tags ;
utf8string lang;
}
}

```

【0433】

同じ情報を複数のグループに複製することを回避するために、エンティティのグループ間でグループプロパティを共有する（したがって、場合によっては、単一の画像、画像のシーケンス、又はその両方をグループ化する）ことが望ましい場合がある。

【0434】

第2の実施形態によれば、汎用のgrouping_type(例えば「case」又は「lgrp」)又は特定のgrouping_type(例えば「aebc」、「albc」...)を有する新しいEntityToGroupBoxは、先に述べたオプションのgroup_uuidパラメータのみを包含してもよい。他の全てのグループプロパティは、次のように記述される。

【0435】

グループプロパティは、EntityToGroupBoxesのパラメータとしてではなく、Box又はFullBoxとして定義される。グループプロパティのボックスタイプは、そのプロパティタイプを指定する。グループプロパティは、記述的又は変換的であり得る。変換グループプロパティは、適用される先行する変換を有するエンティティのグループ内の各項目に適用される（項目に（「ipma」ボックスを介して）関連付けられた変換が最初に適用され、次いで、エンティティのグループに関連付けられた変換（以下で説明する）がグループ内の各項目に適用される）。

【0436】

新しいコンテナボックスGroupPropertiesBox（「gprp」）がMetaBox内に作成される。GroupPropertiesBoxは、任意のグループをグループプロパティの順序付けられたセットと関連付けることを可能にする。このGroupPropertiesBoxは、グループプロパティの暗黙的にインデックス化されたリストを包含するGroupPropertyContainerBox（「gpco」）と、エンティティのグループをグループプロパティに関連付ける1つ以上のGroupPropertyAssociationBox（「gpma」）との2つの部分で構成される。

【0437】

一例では、各GroupPropertyAssociationBoxはgroup_idを増加させることによって順序付けられるものとし、GroupPropertyAssociationBoxボックスのセット内では所与のgroup_idの発生は多くとも1つであるものとする。32ビットのgroup_id値が必要な場合を除きバージョン0を使用するべきであり、同様に、GroupPropertyContainerBox内に127を超えるプロパティがある場合を除き、フラグは0に等しくするべきである。バージョン及びフラグの値の所与の対を有するGroupPropertyAssociationBoxは、多くとも1つしか存在しないことになる。

【0438】

関連付けられた構文は次のとおりである。

```

aligned(8) class GroupProperty(property_type) extends Box(property_type)
{
}

aligned(8) class GroupFullProperty(property_type, version, flags) extends FullBox(property_type, version, flags)
{
}

aligned(8) class GroupPropertyContainerBox extends Box('gpco')
{
}

```

```

Box properties[]; //ボックスを満たすために
//GroupProperty、GroupFullProperty、又はFreeSpaceBoxから
//導出されたボックス
}
aligned(8) class GroupPropertyAssociationBox extends FullBox('gpma' version,
flags)
{
    unsigned int(32) entry_count;
    for(i=0; i < entry_count; i++){
    if (version < 1)
    unsigned int(16) group_id;
    else
    unsigned int(32) group_id;
    unsigned int(8) association_count;
    for (i=0; i < association_count; i++){
    bit(1) essential;
    if (flags & 1)
    unsigned int(15) property_index;
    else
    unsigned int(7) property_index;
    }
    }
    }
Box Type:'gprp'
Container: MetaBox ('meta')
Mandatory: No
Quantity: Zero or one
aligned(8) class GroupPropertiesBox extends Box('gprp') {
GroupPropertyContainerBox property_container;
GroupPropertyAssociationBox association;
}

```

【 0 4 3 9 】

ここで、
group_idは、プロパティが関連付けられているEntityToGroupBoxを識別する。

essentialは、1に設定されている場合には関連付けられたプロパティがそのグループに必須であり、そうでない場合には必須ではないことを示す。

property_indexは、プロパティが関連付けられていない(essentialインジケータも0である)ことを示す0か、又は同じGroupPropertiesBoxに包含されるGroupPropertyContainerBox内の関連付けられたプロパティボックスの1ベースのインデックス(FreeSpaceボックスを含む全てのボックスをカウントする)かのいずれかである。

【 0 4 4 0 】

この第2の実施形態によれば、エンティティのグループのコンテンツの名前、タグ、又は記述などの人間に可読なラベルは、例えば次のような特定のFourCC「gdes」を有する特定のGroupDescriptionPropertyとして定義されてもよい。

```

aligned(8) class GroupDescriptionProperty extends GroupFullProperty('gdes',
version = 0, flags = 0){

```

```

utf8string name;
utf8string description;
utf8string tags ;
utf8string lang;
}

```

【 0 4 4 1 】

ここで、

nameは、エンティティのグループの人間に可読な名前を包含するnull終端U T F - 8文字の文字列である。

descriptionは、エンティティのグループの人間に可読な記述を包含するnull終端U T F - 8文字の文字列である。

tagsは、エンティティのグループに関連するカンマ区切りのタグを包含するnull終端U T F - 8文字の文字列である。

langは、「 e n - U S 」、「 f r - F R 」、「 z h - C N 」などのR F C 5 6 4 6 準拠の言語タグ文字列を包含する文字列で、名前、記述、及びタグに包含されるテキストの言語を表す。langが空の場合、言語はunknown/undefinedである。

【 0 4 4 2 】

この第2の実施形態によれば、異なる言語の代替例を表すために、G r o u p P r o p e r t y A s s o c i a t i o n B o x を介して複数のG r o u p D e s c r i p t i o n P r o p e r t y をグループに関連付けることができる。

【 0 4 4 3 】

あるいは、代替言語を有する複数のG r o u p D e s c r i p t i o n P r o p e r t y をグループに関連付けるために、G r o u p D e s c r i p t i o n P r o p e r t y は次のように代替名、記述、及びタグのリストを包含してもよい。

```

aligned(8) class GroupDescriptionProperty extends GroupFullProperty('gdes',
version = 0, flags = 0){
unsigned int(16) entry_count;
for (i=0; i < entry_count; i++){
utf8string group_name;
utf8string group_description;
utf8string group_tags ;
utf8string lang;
}
}

```

【 0 4 4 4 】

変形例では、人間に可読な各文字列は、より高い柔軟性のために、それに関連付けられた言語を有する個別のグループプロパティとして定義することができる。例えば、「tag」文字列に対して1つのプロパティが、「label」文字列に対して1つのプロパティが、「description」文字列に対して1つのプロパティがあってもよい。

【 0 4 4 5 】

上記の変形例では、各グループプロパティは、上記の説明と同様に、代替の2つのテキスト/言語のリストを包含してもよい。

【 0 4 4 6 】

場合によっては、同じプロパティが項目又はエンティティのグループのいずれかに適用されてもよい。例えば、上記のG r o u p D e s c r i p t i o n P r o p e r t y ボックスは、項目又はエンティティのグループのいずれかに人間が表示可能な記述を提供するために有用であり得る。同様に、項目プロパティは、全体として項目又は項目のグループに適用されてもよい。

【 0 4 4 7 】

変形例では、グループプロパティをエンティティのグループに関連付けるために新しいボ

10

20

30

40

50

ックスを定義するのではなく、グループプロパティは項目プロパティボックス (ItemProperty 又は ItemFullProperty) として定義され、ItemPropertyAssociationBox のセマンティックは項目又はエンティティのグループを参照できるように変更される。

【0448】

例えば、ItemPropertyAssociationBox は、一連の項目又はトラックをグループ化するグループの識別子 (例えば、EntityToGroup の group_id パラメータ) を参照することを可能にする。次に、「ipma」ボックスの item_ID 欄は、項目の識別子 (item_ID) 又はグループ識別子 (group_id) のいずれかを参照できる item_or_group_ID に置き換えられる。この変形例の利点は、エンティティのグループ及び項目が同じプロパティを共有することができ、プロパティ定義及び関連付けの繰り返しが回避されるので、プロパティの記述がよりコンパクトになることである。

10

【0449】

以下は、ItemPropertyAssociationBox の新しい構文の例である。

```
aligned(8) class ItemPropertyAssociationBox extends FullBox('ipma' version, flags)
```

```
{
  unsigned int(32) entry_count;
  for(i=0; i < entry_count; i++){
    if (version < 1)
      unsigned int(16) item_or_group_ID;
    else
      unsigned int(32) item_or_group_ID;
    unsigned int(8) association_count;
    for (i=0; i < association_count i++){
      bit(1) essential;
      if (flags & 1)
        unsigned int(15) property_index;
      else
        unsigned int(7) property_index;
    }
  }
}
```

20

30

【0450】

ItemPropertyAssociationBox の異なるパラメータのセマンティックスは、item_or_group_ID に名前変更される item_ID 欄を除いて、変更されないままである。この item_or_group_ID パラメータは、1 つの項目又は 1 つの EntityToGroup のいずれかを参照してもよい。したがって、item_or_group_ID の値は、1 つの group_id 値又は 1 つの item_ID 値に等しくなるはずである。なお、標準における group_id の定義により、group_id は、他の group_id、item_id、又は track_id と等しくはなることはないことには留意されたい。したがって、item_or_group_ID が与えられたならば、項目とグループとのどちらを参照するかがあいまいになるはずがない。

40

【0451】

代替例として、ItemPropertyAssociationBox の既存のバージョン 0 及び 1 との後方互換性を維持するために、ItemPropertyAssociationBox の新しいバージョン 2 及び 3 は、次のように提案及び定義されてもよい。

```
aligned(8) class ItemPropertyAssociationBox extends FullBox('ipma' version, flags)
```

```
{
```

50

```

unsigned int(32) entry_count;
for(i=0; i < entry_count; i++){
  if (version == 0)
    unsigned int(16) item_ID;
  else if (version == 1)
    unsigned int(32) item_ID;
  else if (version == 2)
    unsigned int(16) group_id;
  else
    unsigned int(32) group_id;
  unsigned int(8) association_count;
  for (i=0; i < association_count; i++){
    bit(1) essential;
    if (flags & 1)
      unsigned int(15) property_index;
    else
      unsigned int(7) property_index;
  }
}
}

```

10

20

【 0 4 5 2 】

各 `ItemPropertyAssociationBox` は `item_ID` 又は `group_id` を増加させることによって順序付けられることになり、`ItemPropertyAssociationBox` ボックスのセット内では所与の `item_ID` 又は `group_id` の発生は多くとも1つとなる。32ビット `item_ID` 値が必要な場合を除き、このバージョン0はプロパティを項目に関連付けるために使用されるべきであり、そのような場合にはバージョン1が使用されるべきである。

【 0 4 5 3 】

同様に、32ビットの `group_id` 値が必要な場合を除き、このバージョン2はプロパティをエンティティのグループに関連付けるために使用されるべきであり、そのような場合にはバージョン3が使用されるべきである。

30

【 0 4 5 4 】

`ItemPropertyContainerBox` に127を超えるプロパティがある場合を除き、フラグは0に等しくなければならない。バージョン及びフラグの値の所与の対を有する `ItemPropertyAssociationBox` は、多くとも1つとなる。

【 0 4 5 5 】

別の変形例では、既存の `ItemPropertyAssociationBox` の新しいバージョンを作成するのではなく、新しいボックス `EntityToGroupAssociationBox` 「epma」が、例えば次のように `ItemPropertiesBox` 内に直接作成される。

40

```

aligned(8) class EntityToGroupAssociationBox extends FullBox('epma' version,
flags)
{
  unsigned int(32) entry_count;
  for(i=0; i < entry_count; i++){
    if (version == 0)
      unsigned int(16) group_id;
    else
      unsigned int(32) group_id;
  }
}

```

50

```

unsigned int(8) association_count;
for (i=0; i < association_count; i++){
    bit(1) essential;
    if (flags & 0x1 == 0x1)
        unsigned int(15) property_index;
    else
        unsigned int(7) property_index;
}
}
}

```

10

```

Box Type:'iprp'
Container: MetaBox ('meta')
Mandatory: No
Quantity: Zero or one
aligned(8) class ItemPropertiesBox extends Box('iprp') {
    ItemPropertyContainerBox property_container;
    ItemPropertyAssociationBox association[];
    EntityToGroupAssociationBox groupAssociation[];
}

```

【 0 4 5 6 】

20

各 EntityToGroupAssociationBox は group_id を増加させることによって順序付けされることになり、EntityToGroupAssociationBox ボックスのセットの内には、所与の group_id の発生は多くとも 1 つとなる。32 ビットの group_id 値が必要な場合を除き、このバージョン 0 はプロパティを EntityToGroups に関連付けるために使用されるべきであり、そのような場合にはバージョン 1 が使用されるべきである。

【 0 4 5 7 】

ItemPropertyContainerBox に 127 を超えるプロパティがある場合を除き、フラグは 0 に等しくなければならない。バージョン及びフラグの値の所定の対を有する EntityToGroupAssociationBox は、多くとも 1 つとなる。

30

【 0 4 5 8 】

これらの変形例では、名前、タグ、又は記述などの人間に可読なラベルは、項目又はエンティティのグループのいずれかに適用されてもよい。このようなラベルは、例えば次のような特定の FourCC「ides」を有する特定の記述項目プロパティ ItemDescriptionProperty として定義される。

```

aligned(8) class ItemDescriptionProperty extends ItemFullProperty('ides', version = 0, flags = 0){
    utf8string name;
    utf8string description;
    utf8string tags ;
    utf8string lang;
}

```

40

【 0 4 5 9 】

ここで、

name は、項目又はエンティティのグループの人間に可読な名前を包含する null 終端 UTF - 8 文字の文字列である。

description は、項目又はエンティティのグループの人間に可読な記述を包含する null 終端 UTF - 8 文字の文字列である。

tags は、項目又はエンティティのグループに関連するカンマ区切りのタグを包含する null

50

終端UTF-8文字の文字列である。

langは、「en-US」、「fr-FR」、「zh-CN」などのRFC5646準拠の言語タグ文字列を包含する文字列で、名前、記述、及びタグに包含されるテキストの言語を表す。langが空の場合、言語はunknown/undefinedである。

【0460】

変形例では、人間に可読な各文字列は、より高い柔軟性のために、それに関連付けられた言語を有する個別の記述項目プロパティとして定義することができる。例えば、「tag」文字列に対して1つのプロパティが、「label」文字列に対して1つのプロパティが、「description」文字列に対して1つのプロパティがあってもよい。

【0461】

項目プロパティは順序付けされ、標準で記述されているように、項目プロパティは記述項目プロパティ又は変換項目プロパティのいずれかになる可能性がある。変換項目プロパティの例は、画像回転又は画像切り抜きである。記述項目プロパティの例は、色情報又は画素情報である。

【0462】

上記の実施形態によれば、記述項目プロパティがエンティティのグループに関連付けられる場合、それは、項目プロパティのセマンティックに応じて、グループ内の各エンティティに適用される共通のプロパティを記述するか、又は全体としてエンティティのグループに適用される。あるいは、「ipma」ボックス（又は、実施形態に応じて「epma」ボックス）のボックスフラグ値は、そのボックスによって記述された記述的な項目プロパティの関連付けがグループ全体又はグループ内の各項目に適用される場合、シグナリングを可能にする。変換項目プロパティは、エンティティのグループに関連付けられる場合には、適用される先行する変換を有するグループ内の各エンティティに適用され、すなわち、まず（「ipma」ボックスを介して）項目に関連付けられた変換が適用され、次いで（上述のように）エンティティのグループに関連付けられた変換がグループ内の各項目に順に適用される。

【0463】

前の実施形態によれば、1つ以上の項目及び/又はグループプロパティを、項目のプロパティ自体又はグループ全体に適用される共通プロパティのいずれかを記述するために、各項目又はエンティティのグループに関連付けることができる。

【0464】

しかしながら、制限された範囲内でプロパティをエンティティに関連付けることができること、すなわち、所与のエンティティのグループのコンテキストにおいてのみ適用されるであろうエンティティにプロパティを関連付けることができることも望ましい場合がある。

【0465】

例えば、同じ画像が2つの異なるアルバムコレクションに属し、ユーザが各アルバムコレクションの範囲内でこの画像を記述する人間が表示可能な異なるテキストを関連付けることを望む場合がある。例えば、山の前の車とともに風景を表す画像は、車専用のものと休日専用の別のものととの2つの異なるアルバムコレクションと関連付けられてもよい。第1のアルバムコレクションではユーザがテキスト「素敵な赤い車！」を画像に関連付けたい場合があり、一方、第2のアルバムコレクションではユーザがテキスト「山での私の素敵な休日」を関連付けたい場合がある。

【0466】

別の例として、任意のコレクションの外側で、画像は、HEIFファイル内で、画像の切り抜きを実現する変換項目プロパティクリーン口径と関連付けられてもよい。しかし、ユーザは、異なる切り抜きパラメータを有する同じファイル内の異なるコレクションにこの画像を追加したい。

【0467】

上記の両方のユースケースにおいて、特定のグループのコンテキストにおける画像にプロパティを関連付けることが有用である。

10

20

30

40

50

【 0 4 6 8 】

一実施形態では、グループに属する項目を考慮するとき、この項目は、この項目に無条件に関連付けられた（グループの範囲内にないことを意味する）全てのプロパティ、及びそのグループの範囲内でこの項目に関連付けられたプロパティに関連付けられる。同じプロパティが無条件に、そのグループの範囲内で定義されている場合、グループの範囲内で定義されているプロパティが優先権を有し、その値が無条件プロパティに定義されている値を上書きする。

【 0 4 6 9 】

第3の実施形態によれば、第1の実施形態で導入された、撮像シリーズ又はコレクショングループを記述する新しい `EntityToGroupBox` の、汎用的なもの（例として、`CaptureSeriesEntityToGroupBox` 又は `LogicalEntityToGroupBox`）、又は特定のもの（例として、`FavouriteCollectionEntityToGroupBox`、`AlbumCollectionEntityToGroupBox` 又は `AutoExposureBracketingEntityToGroupBox`）のいずれかは、以下の例に示すように、プロパティインデックスのリストをグループ内の各エンティティに関連付けるように拡張される。

```
aligned(8) class CaptureSeriesEntityToGroupBox extends EntityToGroupBox('case', version, flags){
    unsigned int(32) capture_mode;
    for(i = 0; i < num_entities_in_group; i++){
        unsigned int(32) entity_id;
        unsigned int(8) association_count;
        for(i=0; i < association_count; i++){
            bit(1) essential;
            if (flags & 0x1 == 0x1)
                unsigned int(15) property_index;
            else
                unsigned int(7) property_index;
        }
    }
}
```

【 0 4 7 0 】

ここで、

`capture_mode`は撮像又はコレクションモードを識別する。

`num_entities_in_group`は `EntityToGroupBox` から継承され、グループ内のエンティティ（項目又はトラック）の数を表す。

`entity_id`は、プロパティが関連付けられているエンティティを識別する。

`essential`は、1に設定されている場合には関連付けられたプロパティがこの項目に必須であり、そうでない場合には必須でないことを示す。

`property_index`は、プロパティが関連付けられていない（`essential`インジケータも0である）ことを示す0か、又は囲んでいる `GroupListBox` と同じレベルに存在する `ItemPropertiesBox` に包含される `ItemPropertyContainerBox` 内の関連付けられたプロパティボックスの1ベースのインデックス（`FreeSpace` ボックスを含む全てのボックスをカウントする）である。

【 0 4 7 1 】

同様の例及びパラメータのセマンティクスは、次のように、撮像又はコレクションモードの種類ごとに特定の `grouping_type` を有する `EntityToGroupBox` のケースを示す。

```
aligned(8) class AlbumCollectionEntityToGroupBox extends EntityToGroupBox
```

```

('albc', version, flags) {
for(i = 0; i    num_entities_in_group; i++){
unsigned int(32) entity_id;
unsigned int(8) association_count;
for (i=0; i    association_count; i++){
bit(1) essential;
if (flags & 0x1 == 0x1)
unsigned int(15) property_index;
else
unsigned int(7) property_index;
}
}
}

```

10

【 0 4 7 2 】

図 5 に示される変形例では、グループ 5 2 0 の範囲内の項目 5 1 0 を有する項目プロパティ 5 4 0 の関連付けは、MetaBox 5 0 の ItemPropertiesBox 内で直接行われる。新しいボックス 5 3 3 は、次のように ItemPropertiesBox 内で定義される。

```

aligned(8) class ItemPropertyInGroupAssociationBox extends FullBox('gpma'v
ersion, flags)

```

20

```

{
unsigned int(32) group_entry_count;
for(i = 0; i    group_entry_count; i++){
if (version    1)
unsigned int(16) group_id;
else
unsigned int(32) group_id;
unsigned int(32) entry_count;
for(i = 0; i    entry_count; i++){
if (version    1)
unsigned int(16) item_ID;
else
unsigned int(32) item_ID;
unsigned int(8) association_count;
for (i = 0; i    association_count; i++){
bit(1) essential;
if (flags & 0x1 == 0x1)
unsigned int(15) property_index;
else
unsigned int(7) property_index;
}
}
}

```

30

Box Type: 'iprp'

Container: MetaBox ('meta')

Mandatory: No

Quantity: Zero or one

```

aligned(8) class ItemPropertiesBox extends Box('iprp') {
ItemPropertyContainerBox property_container;
ItemPropertyAssociationBox association[];

```

40

50

```
ItemPropertyInGroupAssociationBox ItemInGroupAssociation[];
}
```

【 0 4 7 3 】

ここで、

group_entry_countは、グループのリスト内のエントリの数を提供する。

group_idは、プロパティが関連付けられるコンテキストを定義するグループを識別する。

entry_countは、グループ内の項目のリスト内の項目の数を提供する。

item_IDは、プロパティが関連付けられる項目を識別する。

essentialは、1に設定されている場合には、関連付けられたプロパティがその項目に必須であり、そうでない場合には必須ではないことを示す。

property_indexは、プロパティが関連付けられていない (essentialインジケータも0である) ことを示す0か、又は同じItemPropertyContainerBoxに包含されるItemPropertyContainerBox内の関連付けられたプロパティボックスの1ベースのインデックス (FreeSpaceボックスを含む全てのボックスをカウントする) かのいずれかである。

【 0 4 7 4 】

変形例では、EntityToGroupBoxのgroup_id == group_idを有する関連付けられたEntityToGroupBox内のitem_ID又はtrack_IDのいずれかを指定するために、item_IDがentity_IDに置き換えられてもよい。

【 0 4 7 5 】

各ItemPropertyInGroupAssociationBoxはgroup_id及びitem_IDを増加させることによって順序付けられることになり、ItemPropertyInGroupAssociationBoxボックスのセット内には所与のgroup_idの発生が多くとも1つとなる。バージョン0は32ビットitem_ID値が必要とされる場合を除き使用されるべきであり、同様に、ItemPropertyContainerBox内に127を超えるプロパティがある場合を除き、フラグは0に等しくなるべきである。バージョン及びフラグの値の所与の対を有するItemPropertyAssociationBoxは、多くとも1つとなる。

【 0 4 7 6 】

別の変形例では、ItemPropertiesBoxで新しいボックスを定義するのではなく、次のように、既存のItemPropertyAssociationBox(「ipma」)が新しいバージョン2を用いて拡張されてもよい。

```
aligned(8) class ItemPropertyAssociationBox extends FullBox('ipma' version, flags)
```

```
{
```

```
if (version == 2)
```

```
    unsigned int(32) group_id;
```

```
    unsigned int(32) entry_count;
```

```
    for(i = 0; i < entry_count; i++){
```

```
        if (version < 1)
```

```
            unsigned int(16) item_ID;
```

```
        else
```

```
            unsigned int(32) item_ID;
```

```
            unsigned int(8) association_count;
```

```
            for (i=0; i < association_count; i++){
```

```
                bit(1) essential;
```

```
                if (flags & 0x1 == 0x1)
```

```
                    unsigned int(15) property_index;
```

```
                else
```

```
                    unsigned int(7) property_index;
```

10

20

30

40

50

```

}
}
}

```

【0477】

ここで、バージョンが2の場合、新しい属性group_idは、item_IDとproperty_indexとの間の関連付けの範囲を、同じgroup_id値を有するEntityToGroupBoxのコンテキストに制限する。

【0478】

各ItemPropertyAssociationBoxは、item_ID(及び存在する場合にはgroup_id)を増加させることによって順序付けられることになる。

10

【0479】

ItemPropertyAssociationBoxボックスのセット内には、所与のgroup_idが多くとも1つとなる。

【0480】

バージョン0及び1を有するItemPropertyAssociationBoxボックスのセット内には、所与のitem_IDの発生は多くとも1つとなる。

【0481】

ItemPropertyContainerBoxに127を超えるプロパティがある場合を除き、1stビット(LSB)フラグは0に等しくなければならない。

【0482】

20

バージョン及びバージョン0又は1を有するフラグの値の所与のペアを有するItemPropertyAssociationBoxは、多くとも1つとなる。

【0483】

HEIF標準では、項目プロパティは順序付けされ、項目プロパティは記述項目プロパティ又は変換項目プロパティのいずれかになる可能性がある。いくつかの実施形態によれば、同じタイプを有する複数の項目プロパティを、グローバルに、又は特定のグループの範囲内で、のいずれかで、同じ項目に関連付けることができる。例えば、(タイプ「iro」の)変換項目プロパティ画像回転は、バージョン0又は1を有するItemPropertyAssociationBoxを使用して、汎用の範囲内の項目に関連付けられてもよい。同時に、この同じ項目は、例えばバージョン2を有するItemPropertyAssociationBoxを使用して、所与のグループの限定された範囲内の(タイプ「iro」の)別の項目プロパティ画像回転に関連付けられてもよい。

30

【0484】

そのような場合、いくつかの実施形態によれば、変換項目プロパティは適用された先行する変換を有する項目に適用され、所与のグループの範囲内の項目の場合、汎用の範囲内の変換項目プロパティはこの所与のグループの範囲内の変換項目プロパティの前に適用される。逆に、記述項目プロパティの場合、グループの範囲内の項目に関連付けられた記述項目プロパティは、汎用の範囲内の同じ項目に関連付けられた同じタイプを有する記述項目プロパティに取って代わる。

【0485】

40

あるいは、追加の1ビット属性supersede_in_group_flagが、グループの範囲内で関連付けられた項目プロパティが汎用の範囲内の同じ項目に関連付けられた同じタイプを有する項目プロパティに取って代わるかそうでないかをシグナリングするために、ItemPropertyAssociationBoxの新しいバージョンに追加される。

【0486】

この追加の属性を有するItemPropertyAssociationBoxの構文の例は以下に記述される。

```
aligned(8) class ItemPropertyAssociationBox extends FullBox('ipma', version, flags)
```

```
{
```

50

```

if (version == 2)
unsigned int(32) group_id;
unsigned int(32) entry_count;
for(i = 0; i < entry_count; i++){
if (version == 1)
unsigned int(16) item_ID;
else
unsigned int(32) item_ID;
unsigned int(8) association_count;
for (i=0; i < association_count; i++){
bit(1) essential;
if (version == 2){
bit(1) supersede_in_group_flag;
if (flags & 0x1 == 0x1)
unsigned int(14) property_index;
else
unsigned int(6) property_index;
}
else{
if (flags & 0x1 == 0x1)
unsigned int(15) property_index;
else
unsigned int(7) property_index;
}
}
}
}
}

```

10

20

【 0 4 8 7 】

ここで、バージョンが 2 に等しい場合、新しい属性group_idはitem_IDとproperty_indexとの間の関連付けの範囲を、同じgroup_id値を有するEntityToGroupBoxのコンテキストに制限する。さらに、1 に設定された場合の新しい1 ビット属性(supersede_in_group_flag) は、関連付けられたプロパティが汎用の範囲で同じタイプを有するプロパティに取って代わることを示し、もしそうでない場合には、関連付けられたプロパティは汎用の範囲で同じ項目に関連付けられた同じタイプを有するプロパティの後の順に適用される。

30

【 0 4 8 8 】

あるいは、このsupersede_in_group_flag値は、ItemPropertyAssociationBoxのフラグパラメータで特定のフラグ値としてシグナリングされる場合がある。このような場合、supersede_in_group_flag値は、ItemPropertyAssociationBoxで宣言された全てのプロパティ関連付けに適用される。

40

【 0 4 8 9 】

第 4 の実施形態によれば、新しいフラグを有するItemPropertyAssociationBox(「ipma」)の新しいバージョン(バージョン = 2 及びバージョン = 3) は、項目、エンティティのグループ、又はエンティティのグループ内の項目のいずれかとの項目プロパティの全ての可能な関連付けを提供する、この単一メタデータ構造の全てのバージョンを扱うように定義される。

【 0 4 9 0 】

図 6 は、本発明のいくつかの実施形態による、項目プロパティと項目との間の異なる関連付け、エンティティのグループ、又はエンティティのグループの範囲内の項目を示す。

【 0 4 9 1 】

50

項目プロパティ 1 と項目 1 との間の関連 6 0 1 は、項目プロパティの項目との関連付けを示す。この関連付けは、そのファイルの範囲内で有効であり、これは汎用の関連付けである。

【 0 4 9 2 】

項目プロパティ 3 と、項目 3 及び項目 4 によって構成される項目のグループとの間の関連 6 0 2 は、項目プロパティの項目のグループとの関連付けを示す。

【 0 4 9 3 】

項目 4 と項目プロパティ 5 との間の関連 6 0 3 は、項目プロパティの、項目 3 と項目 4 とからなる項目のグループの範囲内の項目との関連付けを示す。

【 0 4 9 4 】

項目 4 は、その項目のグループへの所属とは無関係に、項目プロパティ 1 及び項目プロパティ 2 に無条件に関連付けられることに留意されたい。

【 0 4 9 5 】

この実施形態によれば、ItemPropertyAssociationBox は、例えば、以下のように定義されてもよい。

次のフラグは、「ipma」ボックスフラグに設定されることが可能である。

- 0 x 0 0 0 0 0 1 は、property_index属性のサイズを示す。

o このビットが設定されている場合、property_index_size = 1 5。

o それ以外の場合、property_index_size = 7。

- 0 x 0 0 0 0 0 2 は、item_ID及びgroup_id属性のサイズを示す。

o このビットが設定されている場合、item_or_group_ID_size = 3 2。

o それ以外の場合、item_or_group_ID_size = 1 6。

【 0 4 9 6 】

```
aligned(8) class ItemPropertyAssociationBox extends FullBox('ipma' version, flags)
```

```
{
```

```
if (version == 2)
```

```
    unsigned int(item_or_group_ID_size) context_group_id;
```

```
    unsigned int(32) entry_count;
```

```
    for(i = 0; i < entry_count; i++){
```

```
        if (version == 1)
```

```
            unsigned int(16) item_ID;
```

```
        else if (version == 1)
```

```
            unsigned int(32) item_ID;
```

```
        else if (version == 2)
```

```
            unsigned int(item_or_group_ID_size) item_ID;
```

```
        else
```

```
            unsigned int(item_or_group_ID_size) group_id;
```

```
            unsigned int(8) association_count;
```

```
            for (i=0; i < association_count; i++){
```

```
                bit(1) essential;
```

```
                if (version == 2){
```

```
                    bit(1) supersede_in_group_flag;
```

```
                    unsigned int(property_index_size) property_index;
```

```
                }
```

```
            else
```

```
                unsigned int(property_index_size) property_index;
```

```
            }
```

```
        }
```

```
    }
```

```
}
```

10

20

30

40

50

【 0 4 9 7 】

ここで、

context_group_idは、item_IDとproperty_indexとの間の関連付けの範囲を、同じgroup_id値を有するEntityToGroupBoxのコンテキストに制限する。

group_idは、プロパティが関連付けられているEntityToGroupBoxを識別する。

essentialは、1に設定されている場合には関連付けられたプロパティがそのグループに必須であり、そうでない場合には必須ではないことを示す。

1に設定された場合のsupersede_in_group_flagは、関連付けられたプロパティが汎用の範囲で同じタイプを有するプロパティに取って代わることを示し、もしそうでない場合には、関連付けられたプロパティは汎用の範囲で同じ項目に関連付けられた同じタイプを有するプロパティの後の順に適用される。

property_indexは、プロパティが関連付けられていない(essentialインジケータも0である)ことを示す0か、又は同じGroupPropertiesBoxに包含されるGroupPropertyContainerBox内の関連付けられたプロパティボックスの1ベースのインデックス(FreeSpaceボックスを含む全てのボックスをカウントする)かのいずれかである。

【 0 4 9 8 】

「ipma」ボックスのバージョン2と3との定義は後方互換であり、「ipma」ボックスの既存のバージョン0と1とを変更しない。

【 0 4 9 9 】

各ItemPropertyAssociationBoxは、item_ID又はgroup_id(及び存在する場合にはcontext_group_id)を増加させることによって順序付けられることになる。

【 0 5 0 0 】

ItemPropertyAssociationBoxボックスのセット内には、所与のcontext_group_idが多くとも1つとなる。

【 0 5 0 1 】

バージョン0、1又は3を有するItemPropertyAssociationBoxボックスのセットには、所与のitem_ID及びgroup_idの発生は多くとも1つとなる。

【 0 5 0 2 】

ItemPropertyContainerBoxに127を超えるプロパティがある場合を除き、1stビット(LSB)フラグは0に等しくなければならない。

【 0 5 0 3 】

バージョン = 2を除いた、バージョンと値0x1又は0x2を有するフラグとの所与の値の対を有するItemPropertyAssociationBoxは多くとも1つとなる。

【 0 5 0 4 】

この先の実施形態によれば、ItemPropertyAssociationBoxは、例えば、次のように定義されてもよい。

次のフラグは、「ipma」ボックスフラグに設定されることが可能である。

- 0x0000001は、property_index属性のサイズを示す。

oこのビットが設定されている場合、property_index_size = 15。

oそれ以外の場合、property_index_size = 7。

- 0x0000002は、はitem_ID及びgroup_id属性のサイズを示す。

oこのビットが設定されている場合、item_or_group_ID_size = 32。

oそれ以外の場合、item_or_group_ID_size = 16。

- 0x0000004は、group_limited_scope。

oこのビットが設定されている場合、それは、このボックスで宣言された関連付けが属性context_group_idで識別されるグループの範囲に制限されていることを示す。

10

20

30

40

50

【 0 5 0 5 】

```

aligned(8) class ItemPropertyAssociationBox extends FullBox('ipma' version, flags)
{
  if (version == 1 && group_limited_scope)
    unsigned int(item_or_group_ID_size) context_group_id;
  unsigned int(32) entry_count;
  for(i = 0; i < entry_count; i++){
    if (version == 1)
      unsigned int(16) item_ID;
    else if (version == 1)
      unsigned int(32) item_ID;
    else if (version == 1 && group_limited_scope)
      unsigned int(item_or_group_ID_size) item_ID;
    else
      unsigned int(item_or_group_ID_size) group_id;
    unsigned int(8) association_count;
    for (i=0; i < association_count; i++){
      bit(1) essential;
      if (version == 1 && group_limited_scope){
        bit(1) supersede_in_group_flag;
        unsigned int(property_index_size-1) property_index;
      }
      else
        unsigned int(property_index_size) property_index;
    }
  }
}

```

10

20

【 0 5 0 6 】

ここで、
 context_group_idは、item_IDとproperty_indexとの間の関連付けの範囲を、同じgroup_id値を有するEntityToGroupBoxのコンテキストに制限する。
 group_idは、プロパティが関連付けられているEntityToGroupBoxを識別する。

30

essentialは、1に設定されている場合には関連付けられたプロパティがそのグループに必須であり、そうでない場合には必須ではないことを示す。

1に設定された場合のsupersede_in_group_flagは、関連付けられたプロパティが汎用の範囲で同じタイプを有するプロパティに取って代わることを示し、もしそうでない場合には、関連付けられたプロパティは汎用の範囲で同じ項目に関連付けられた同じタイプを有するプロパティの後の順に適用される。

40

property_indexは、プロパティが関連付けられていない(essentialインジケータも0である)ことを示す0か、又は同じGroupPropertiesBoxに包含されるGroupPropertyContainerBox内の関連付けられたプロパティボックスの1ベースのインデックス(FreeSpaceボックスを含む全てのボックスをカウントする)かのいずれかである。

【 0 5 0 7 】

「ipma」ボックスのバージョン2の定義は後方互換であり、「ipma」ボックスの既存のバージョン0と1とを変更しない。

【 0 5 0 8 】

各ItemPropertyAssociationBoxは、item_ID又はgroup_id(

50

及び存在する場合にはcontext_group_id)を増加させることによって順序付けられることになる。

【0509】

ItemPropertyAssociationBoxボックスのセット内には、所与のcontext_group_idが多くとも1つとなる。

【0510】

group_limited_scopeが設定されていないItemPropertyAssociationBoxボックスのセットには、所与のitem_IDとgroup_idとの発生は多くとも1つとなる。

【0511】

ItemPropertyContainerBoxに127を超えるプロパティがある場合を除き、1stビット(LSB)フラグは0に等しくなければならない。

【0512】

フラグgroup_limited_scopeが存在するボックスを除いて、バージョンとフラグとの値の所与の対を有するItemPropertyAssociationBoxは多くとも1つとなる。

【0513】

図7は、HEIFフォーマットを使用して1つ以上のエンティティを1つのファイルにカプセル化する処理の主なステップを示す。計算装置900(図9)は例えば、この処理を適用してもよい。

【0514】

まず、ステップ701で、1つ以上のエンティティが取得される。それらは、上述したような撮像装置による撮像モードに従って取得され得る。

【0515】

代替例では、ステップ701で取得された1つ以上のエンティティを、既存のエンティティ又は以前に撮像されたエンティティ又は作成されたエンティティの中から選択することができる。例えば、既存のエンティティは、既存のHEIFファイル、又はその他のタイプのファイル、例えばJPEG、GIF、BMP、MKV、MP4、AVIから抽出できる。

【0516】

ステップ702で、エンティティの1つ以上の新しいグループが作成される。これらのグループは撮像時に選択された撮像モードから生じてもよく、すなわち、新しいグループは、画像の撮像中に撮像モードを適用することから生じるエンティティの各セットに対して作成される。例えば、撮影時には、毎秒の画像を包含するタイムラプスが実現される。タイムラプス画像は、撮像の終わりに同じグループ内に一緒にグループ化される。別の例として、自動露光ブラケット撮像から生じる全ての画像を一緒にグループ化するために、自動露光ブラケットグループが作成される。エンティティの新しいグループは、自動処理によって、又はユーザ若しくはHEIFファイルの作成者によって行われる編集操作中に作成されてもよい。例えば、エンティティのグループは、ユーザのコレクション又はユーザのお気に入り、若しくは写真シリーズ、若しくは写真セット、若しくはユーザ定義の撮像シリーズを作成するなど、論理的にグループ化されたエンティティに対して作成され得る。ユーザは例えば、画像の集合を形成するために、いくつかのエンティティをグループ化してもよい。例えば自動処理は、同じ場所に、及び/又は予め定められた時間隔内に、写真ショットをグループ化してもよい。例えば撮像装置は、撮像画像の(例えば、GPSセンサからの)位置情報を使用して、その位置に対応する町の名前を決定する。同じ町で撮られた全ての画像は、画像の1つのグループを形成する。例えば、撮像又は編集装置は、画像を分類し、それらを一緒にグループ化するために、人工知能アルゴリズム(例えば、顔又は物体認識)を使用してもよい。

【0517】

ステップ703で、グループのオブジェクトをより正確に記述するために、いくつかのプ

10

20

30

40

50

ロパティが各グループに関連付けられてもよい。例えばグループは、コンテンツのタグ、ラベル、名前、記述テキスト、位置座標、撮像モードの共通パラメータに関連付けられてもよい。例えば撮像モードの共通パラメータは、タイミング（例えば、連続する画像間の獲得フレームレート又は時間デルタ）、連続する撮像間の撮像パラメータ変動（例えば、露光ステップ）の差、又はブラケットモードのプロパティ（例えば、連続、単一若しくは自動ブラケット、又はパノラマ方向）に関連してもよい。

【0518】

いくつかの実施形態では、オプションのステップ705で、いくつかのプロパティがエンティティのグループに属するエンティティに関連付けられてもよく、この関連付けはグループの範囲内でのみ有効である。これは、そのエンティティがグループの一部であると見なされる場合にのみプロパティがエンティティに適用されることを意味する。エンティティがグループとは独立して考慮される場合、そこではプロパティはエンティティに適用されない。

10

【0519】

最後に、ステップ704で、エンティティの画像及びシーケンスは、それらを記述し、それらの関連付けられたプロパティを有するエンティティの作成されたグループを記述するメタデータ構造を有するHEIFファイルフォーマットでカプセル化される。

【0520】

画像の記憶のために、2つの主な可能代替例が利用可能である。

【0521】

20

第1の代替例では、画像は独立して符号化され、HEIF項目としてファイルに格納される。このカプセル化ステップの間に、撮像の条件に関する追加情報がファイル内に提供されてもよい。例えば、自動露光ブラケットモードの場合、各画像の撮像に使用される露光データが提供されてもよい。この説明は、ItemPropertyボックスのプロパティを使用して提供される。

【0522】

第2の代替例では、画像は「pict」又は「vide」トラックに格納される。追加情報は、SampleEntry又はSampleGroupEntryボックスを使用して提供されてもよい。画像の符号化はビデオ符号化と同様に、HEVCエンコーダを使用して以前の画像に依存してもよい。そのトラック内の以前の画像は、予測符号化のための基準画像として利用可能である。

30

【0523】

画像のグループとしての撮像された画像の格納は、先に述べた利用可能なEntityToGroupグループ化機構を使用して、ファイル内にシグナリングされる。

【0524】

図8は、エンティティのグループに関連付けられたプロパティと、グループの範囲内のエンティティ（項目又はトラック）に関連付けられたプロパティとを決定するために、図7のカプセル化処理によって生成されたHEIFファイルの解析処理の主なステップを示している。デコード処理は、1つ以上の画像（項目）又は画像のシーケンス（「pict」又は「vide」トラック）を有するHEIFファイルの解析によって開始される。ステップ801で、エンティティのグループを決定するために、GroupsListBoxに包含されるグループ化情報が解析される。エンティティの各グループのグループ化タイプはグループ化情報が存在する場合に決定され、すなわち、このHEIFファイルは、先に述べた値の1つに等しいgrouping_typeを有するEntityToGroupBoxを含む。第1の代替例では、grouping_typeパラメータはグループ化モードを直接指定する。第2の代替案では、グループ化情報は、画像のセットが汎用撮像シリーズ又はコレクショングループに所属する（グループ化タイプが「case」、「brak」、「udcs」、又は類似の意味を有する任意の他のFourCCに等しい）ことをシグナリングする。この場合、EntityToGroupBoxの追加の属性は、特定のグループ化モードを提供する。

40

50

【0525】

各項目に関連付けられたプロパティ情報はステップ802で、ItemPropertiesBox(「iprp」)を解析し、特定の項目に関連付けられたItemPropertyContainerBoxから各ItemProperty又はItemFullPropertyを引き出すためにItemPropertyAssociationBoxes(バージョン0又はバージョン1)を使用することによって解析される。

【0526】

エンティティの各グループに関連付けられたプロパティ情報は、ステップ503で、それぞれのEntityToGroupBoxで定義された追加の属性を解析することによって解析される。代替例では、エンティティの各グループに関連付けられたプロパティ情報は、専用ボックスGroupPropertiesBox(「grp」)を解析し、特定のグループに関連付けられたGroupPropertyContainerBoxから各ItemProperty又はItemFullPropertyを引き出すためにGroupPropertyAssociationBoxを使用することによって取得される。別の代替例では、エンティティの各グループに関連付けられているプロパティ情報は、ItemPropertiesBox(「grp」)を解析し、特定のグループに関連付けられているItemPropertyContainerBoxから各ItemProperty又はItemFullPropertyを引き出すために専用のItemPropertyAssociationBox(バージョン>=2)を使用することによって取得される。

10

20

【0527】

いくつかの実施形態では、特定のエンティティグループの範囲内の各項目に関連付けられたプロパティ情報は、オプションのステップ804で、項目プロパティのインデックスを提供する、又はそのグループ内の各項目に関連する項目プロパティを直接提供するそれぞれのEntityToGroupBoxで定義された追加の属性を解析することによって、解析される。代替例では、特定のエンティティグループの範囲内の各項目に関連付けられているプロパティ情報は、ItemPropertiesBox(「grp」)を解析し、特定のグループの範囲内の項目に関連付けられているItemPropertyContainerBoxから各ItemProperty又はItemFullPropertyを引き出すために専用のItemPropertyAssociationBox(バージョン>=2)を使用することによって取得される。

30

【0528】

ステップ805で、デコーダは、項目、エンティティのグループ、及びエンティティのグループ内のエンティティに関連付けられた全てのグループ化情報並びにプロパティ情報をプレーヤに提供する。GUIインターフェースは、関連付けられたそれぞれのプロパティ情報を使用して、画像、画像のシーケンス、及びそのようなエンティティのグループの間をユーザがナビゲートすることを可能にするために、特定のGUI要素を示してもよい。このプロパティ情報から、ユーザはHEIFファイルに包含される画像の一部をレンダリングすることを選択してもよい。

【0529】

例えば、HEIFファイルは、一連のブラケット画像を包含する。そのような場合、このアプリケーションは、異なるブラケット代替例を眺めることを可能にするGUIインターフェースを提供する。一実施形態では、インターフェースは、(そのグループからのプロパティ情報とそのグループの範囲内の各画像のプロパティ情報の両方を含む)各画像に関連付けられた撮像の特性を抽出するために、ItemPropertiesなどのプロパティ情報内に提供される情報を提供する。特に、自動露光ブラケット撮影では、ユーザが適切なショットを選択できるように、各ショットの露光絞りがステップ805で表示される。好ましい露光が選択されると、復号装置は、選択された画像を「一次項目」としてマークするようにHEIFファイルを修正してもよい。

40

【0530】

50

撮像モードがパノラマ画像に対応する場合、デコーダはステップ505において、ユーザがナビゲートできる一連の画像をH E I Fファイルが包含することをプレーヤに通知する。このG U Iインターフェースは、ユーザが空間的構成として画像間をナビゲートできるように、特定のG U I要素を示してもよい。プレーヤは、パターンに適合されたナビゲーションインターフェースを生成するように、（例えば、左から右への）セット画像の撮像のパターンを抽出するために、そのグループのプロパティ情報を解析する。例えば、撮像のパターンが左から右である場合、G U Iインターフェースは、H E I Fファイルの項目間をナビゲーションすることを可能にする水平ナビゲーション矢印を提供する。

【0531】

撮像モードが写真シリーズ又は画像バーストに対応する場合、デコーダはステップ805において、例えば、写真シリーズ又は画像バーストグループの全ての画像間のダイアポラマを開始するようにプレーヤに通知する。一実施形態では、ダイアポラマ内の各画像の表示時間は、画像バーストグループのプロパティ情報内に指定される時間隔の関数である。

【0532】

一実施形態では、プレーヤは、ユーザがフォトシリーズのコンテンツを迅速に識別できるように、ダイアポラマの始めに、又は各画像内の透かしとして、プロパティ情報内に提供されるラベル又は名前情報を表示する。別の実施形態では、ユーザは、フォトシリーズグループの一連の画像から、好ましい画像として1つの画像を選択してもよい。このような場合、好ましい画像は、一次項目としてマークされる。別の実施形態では、ユーザは、フォトシリーズから好ましい画像としていくつかの画像を選択してもよい。このような場合、プレーヤは選択された画像を有する新しいフォトシリーズグループを作成し、同じラベルプロパティ情報を関連付ける。

【0533】

図9は、本発明の1つ以上の実施形態を実施するための計算装置900の概略ブロック図である。計算装置900は、マイクロコンピュータ、ワークステーション、又はライトポータブル装置などの装置であってもよい。コンピューティングデバイス900は、以下に接続された通信バスを備える。すなわち、

- ・C P Uと称される、マイクロプロセッサなどの中央処理装置901。

- ・本発明の実施形態に応じた方法を実装するために必要な変数又はパラメータを記録するために適合されたレジスタと同様に、本発明の実施形態の方法の実行可能コードを格納するための、R A Mと称される、そこでメモリ容量は例えば拡張ポートと接続された任意選択のR A Mによって拡張され得る、ランダムアクセスメモリ902。

- ・R O Mと称される、本発明の実施形態を実施するためのコンピュータプログラムを格納するための読み取り専用メモリ903。

- ・ネットワークインターフェース904は、処理されるデジタルデータが送信される又は受信される通信ネットワークへと典型的には接続される。ネットワークインターフェース904は単一のネットワークインターフェースであってもよいし、異なるネットワークインターフェースのセット（例えば、有線及び無線インターフェース、又は異なる種類の有線若しくは無線インターフェース）から構成されてもよい。データパケットは、送信のためにネットワークインターフェースに書き込まれるか、又はC P U 901内で動作しているソフトウェアアプリケーションの制御の下で受信のためにネットワークインターフェースから読み出される。

- ・ユーザインターフェース905は、ユーザから入力を受信するために、又はユーザに情報を表示するために使用されてもよい。

- ・H Dで示されるハードディスク906は、大容量記憶装置として提供されてもよい。

- ・I / Oモジュール907は、ビデオソース又はディスプレイなどの外部装置から / へのデータの受信 / 送信のために使用されてもよい。

【0534】

実行可能コードは、読み取り専用メモリ903、ハードディスク906、又は例えばディスクのような取り外し可能なデジタル媒体のいずれかに格納することができる。変形例に

10

20

30

40

50

よれば、プログラムの実行可能コードは、実行される前にハードディスク 906 などの通信装置 900 の記憶手段の 1 つに格納されるように、ネットワークインターフェース 904 を介して通信ネットワークによって受信され得る。

【0535】

中央処理装置 901 は、前述の記憶手段の 1 つに格納された本発明の実施形態による 1 つ以上のプログラムの命令又はソフトウェアコードの一部の実行を制御し、指示するように適合され、その命令は、前述の記憶手段の 1 つに格納される。電源投入後、CPU 901 は、ソフトウェアアプリケーションに関するメイン RAM メモリ 902 からの命令を、プログラム ROM 903 又はハードディスク 906 からそれらの命令が読み出された後に実行することができる。このようなソフトウェアアプリケーションは、CPU 901 によつて実行されると、本発明のフローチャートの各ステップを行わせる。

10

【0536】

本発明のアルゴリズムの任意のステップは、PC (「Personal Computer」)、DSP (「Digital Signal Processor」)、又はマイクロコントローラなどのプログラム可能な計算マシンによる命令又はプログラムのセットの実行によって、ソフトウェアにおいて実装されてもよく、そうでなければ、FPGA (「Field-Programmable Gate Array」)、又はASIC (「Application-Specific Integrated Circuit」) などの機械又は専用の構成要素によって、ハードウェアにおいて実装されてもよい。

20

【0537】

以上、本発明を具体的な実施形態に基づいて説明したが、本発明は上記実施形態に限定されるものではなく、本発明の要旨を逸脱しない範囲において種々の修正が可能である。

【0538】

さらに、上述した例示的な実施形態を参照すると、多くのさらなる修正及び変更が当業者へと提案されているが、それは単に例として与えられ、添付のクレームによって単独で決定される本発明の範囲を制限することを意図しない。特に、様々な実施形態からの異なる特徴は、適宜、適切な箇所と交換されてもよい。

【0539】

なお、上述した本発明の各実施形態は単独で実施してもよいし、複数の実施形態の組み合わせとして実施してもよい。また、様々な実施形態からの特徴は、必要な場合、又は単一の実施形態における個々の実施形態からの要素又は特徴の組み合わせが有益である場合に組み合わせることができる。

30

【0540】

本明細書に開示される各特徴 (添付の特許請求の範囲、要約及び図面を含む) は、明確に別段の記載がない限り、同一の、同等の、又は同様の目的を果たす代替的特徴に置き換えられてもよい。したがって、明確に別段の記載がない限り、開示される各特徴は、同等又は同様の機能の汎用のシリーズの一例にすぎない。

【0541】

特許請求の範囲において、単語「含む」は他の要素又はステップを排除するものではなく、不定冠詞「a」又は「an」は複数を排除するものではない。異なる特徴が相互に異なる従属請求項に記載されているという単なる事実は、これらの特徴の組み合わせが有利に使用されることができないことを示すものではない。

40

【図面】

【図 1】

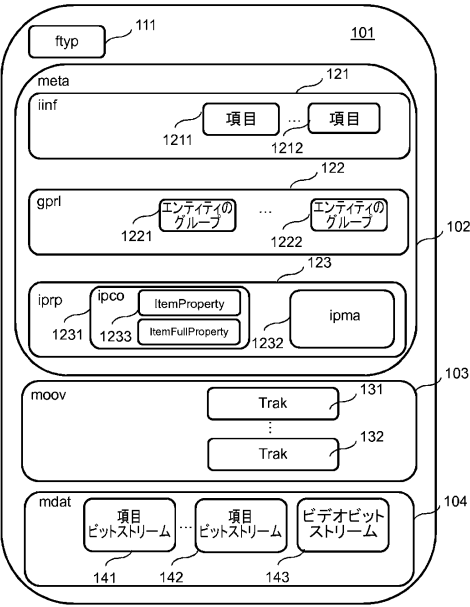


Fig. 1

【図 2】

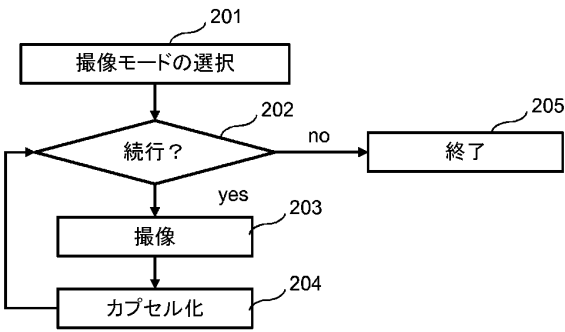


Fig. 2

【図 3】

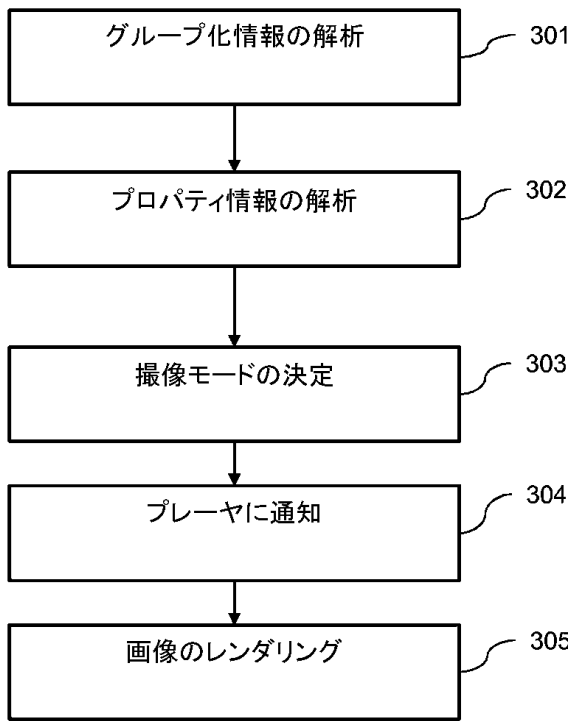


Fig. 3

【図 4】

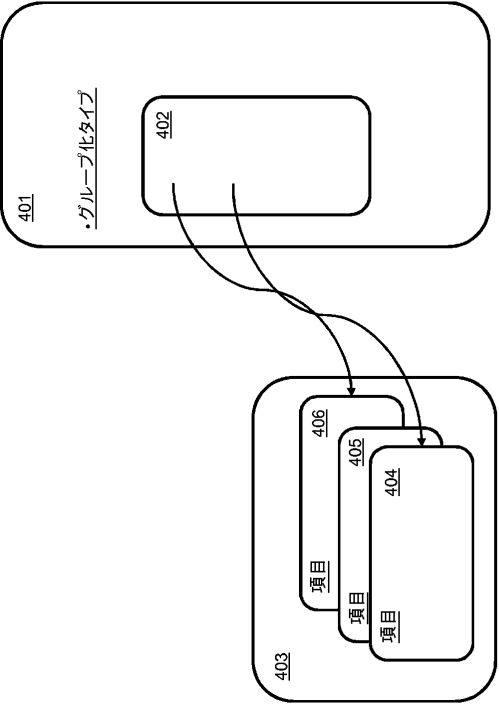


Fig. 4

10

20

30

40

50

【 図 5 】

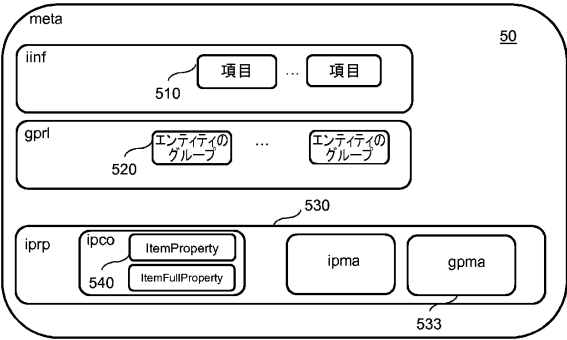


Fig. 5

【 図 6 】

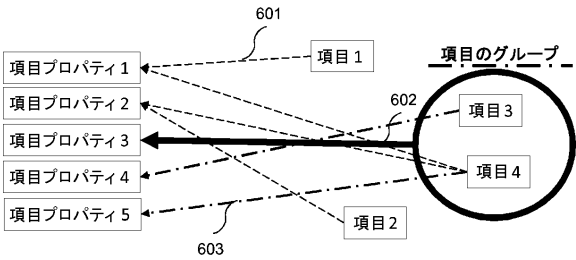


Fig. 6

【 図 7 】

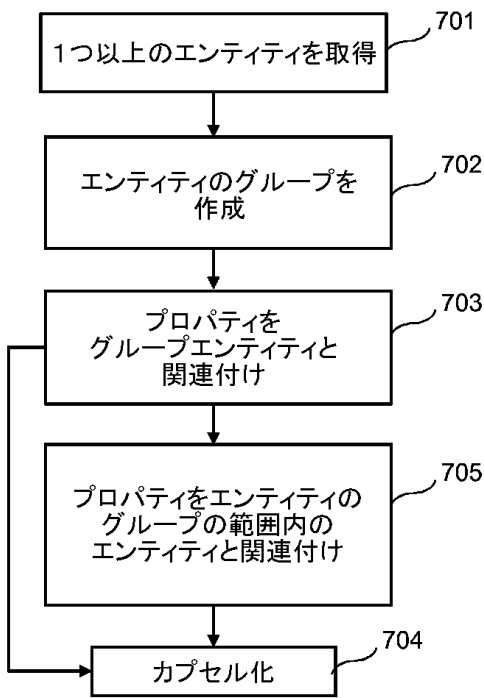


Fig. 7

【 図 8 】

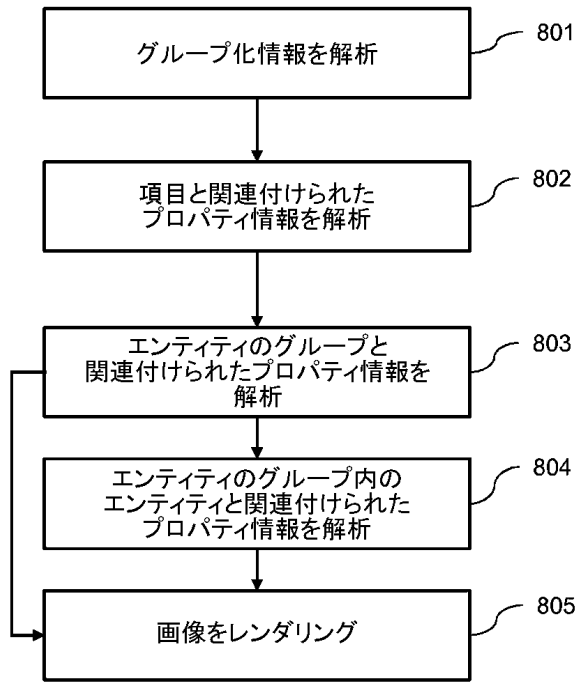


Fig. 8

10

20

30

40

50

【 図 9 】

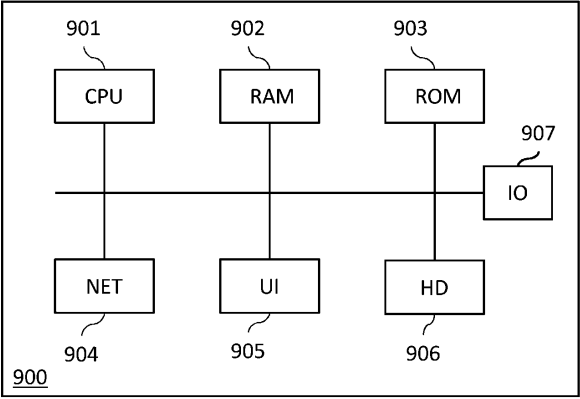


Fig. 9

10

20

30

40

50

フロントページの続き

(33)優先権主張国・地域又は機関

英国(GB)

(31)優先権主張番号 1903174.9

(32)優先日 平成31年3月8日(2019.3.8)

(33)優先権主張国・地域又は機関

英国(GB)

ル キヤノン リサーチ センター フランス エス , エー , エス 内

(72)発明者 ドゥヌアル , フランク

フランス国 レヌ - アタラント , セデックス セッソン - セヴィニエ 3 5 5 1 7 , リュ ドゥ
ラ トゥッシュランベール キヤノン リサーチ センター フランス エス , エー , エス 内

(72)発明者 ルーラン , エルヴェ

フランス国 レヌ - アタラント , セデックス セッソン - セヴィニエ 3 5 5 1 7 , リュ ドゥ
ラ トゥッシュランベール キヤノン リサーチ センター フランス エス , エー , エス 内

(72)発明者 高久 雅彦

神奈川県横浜市緑区上山 3 - 8 - 5

審査官 鈴木 隆夫

(56)参考文献 特開 2 0 0 9 - 2 6 0 9 2 1 (J P , A)

特開 2 0 0 9 - 2 6 6 1 5 7 (J P , A)

国際公開第 2 0 1 6 / 1 2 8 6 1 3 (W O , A 1)

特開 2 0 1 0 - 2 6 7 2 5 6 (J P , A)

特表 2 0 1 8 - 5 3 0 9 6 7 (J P , A)

(58)調査した分野 (Int.Cl. , D B 名)

H 0 4 N 5 / 9 2

H 0 4 N 5 / 9 1 7

H 0 4 N 5 / 2 3 2

H 0 4 N 2 1 / 8 4