

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第6148769号  
(P6148769)

(45) 発行日 平成29年6月14日 (2017. 6. 14)

(24) 登録日 平成29年5月26日 (2017. 5. 26)

(51) Int. Cl.

F I

HO 4 N 5/76 (2006. 01)  
 HO 4 N 5/91 (2006. 01)  
 HO 4 N 5/225 (2006. 01)  
 GO 6 F 17/30 (2006. 01)

HO 4 N 5/76 B  
 HO 4 N 5/76 Z  
 HO 4 N 5/91 J  
 HO 4 N 5/91 Z  
 HO 4 N 5/225 F

請求項の数 15 (全 21 頁) 最終頁に続く

(21) 出願番号 特願2016-125871 (P2016-125871)  
 (22) 出願日 平成28年6月24日 (2016. 6. 24)  
 (65) 公開番号 特開2017-17695 (P2017-17695A)  
 (43) 公開日 平成29年1月19日 (2017. 1. 19)  
 審査請求日 平成28年6月24日 (2016. 6. 24)  
 (31) 優先権主張番号 2015203570  
 (32) 優先日 平成27年6月26日 (2015. 6. 26)  
 (33) 優先権主張国 オーストラリア (AU)

(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 100076428  
 弁理士 大塚 康德  
 (74) 代理人 100115071  
 弁理士 大塚 康弘  
 (74) 代理人 100112508  
 弁理士 高柳 司郎  
 (74) 代理人 100116894  
 弁理士 木村 秀二  
 (74) 代理人 100130409  
 弁理士 下山 治  
 (74) 代理人 100134175  
 弁理士 永川 行光

最終頁に続く

(54) 【発明の名称】 複数のイベントクラスタを生成するために画像セットを分割するための方法、システムおよび装置

(57) 【特許請求の範囲】

【請求項 1】

複数のイベントクラスタであって、各イベントクラスタが画像セットからの少なくとも一つの画像を含む、複数のイベントクラスタを生成するために前記画像セットを分割する方法であって、

前記方法は、

前記セットの各画像に対応する時間メタデータを有する画像にアクセスする工程と、  
 前記画像セットにおける隣接した画像の各ペア間の時間間隔を決定する工程であって、各前記ペアにおける隣接した画像は、時間的に隣接し、各前記ペアに対する前記時間間隔は、前記ペアの各画像に関連する時間メタデータから決定される、工程と、

所定の時間を示す情報とカレンダーの情報の少なくとも一つに基づいて、前記決定された時間間隔の一つ以上に対する調整値を決定する工程と、

前記複数のイベントクラスタを生成するために前記画像セットにおける前記画像を分割する工程であって、各イベントクラスタに対する分割の境界は、少なくとも前記決定された調整値を用いて決定される、工程と、  
 を有することを特徴とする方法。

【請求項 2】

前記調整値は、前記時間間隔と、所定の期間との重なりに基づいて決定されることを特徴とする請求項 1 に記載の方法。

【請求項 3】

前記調整値は、減少されることを特徴とする請求項 2 に記載の方法。

【請求項 4】

前記調整値は、前記時間間隔と、所定の期間とが重なっていない部分に基づいて決定されることを特徴とする請求項 1 に記載の方法。

【請求項 5】

前記調整値は、増大されることを特徴とする請求項 4 に記載の方法。

【請求項 6】

前記所定の期間は、固定であることを特徴とする請求項 2 に記載の方法。

【請求項 7】

重なる期間が週末にかかる場合、前記決定された時間間隔は、さらに減少されることを特徴とする請求項 6 に記載の方法。

10

【請求項 8】

重なる期間が、週末、休暇のシーズン、複数日の祝祭のうちの一つ以上にかかる場合に、前記決定された時間間隔は、さらに減少されることを特徴とする請求項 6 に記載の方法。

【請求項 9】

前記重なる期間が、撮影者の結婚記念日の付近の期間、前記撮影者が休暇をとると予想される期間の一つ以上に関連する場合、前記決定された時間間隔は、さらに減少されることを特徴とする請求項 6 に記載の方法。

【請求項 10】

20

前記画像セットは、撮像を表す前記時間メタデータに基づいて順序付けされることを特徴とする請求項 1 に記載の方法。

【請求項 11】

前記時間間隔は、最初の時間間隔であることを特徴とする請求項 1 に記載の方法。

【請求項 12】

前記調整された時間間隔と前記イベントクラスタに関連するイベントの時間間隔との比較に基づいて、画像をイベントクラスタに割り当てる工程をさらに有することを特徴とする請求項 1 に記載の方法。

【請求項 13】

複数のイベントクラスタであって、各イベントクラスタが画像セットからの少なくとも一つの画像を含む、複数のイベントクラスタを生成するために前記画像セットを分割するための装置であって、

30

前記装置は、

前記セットの各画像に対応する時間メタデータを有する画像にアクセスする手段と、

前記画像セットにおける隣接した画像の各ペア間の時間間隔を決定する手段であって、各前記ペアにおける隣接した画像は、時間的に隣接し、各前記ペアに対する前記時間間隔は、前記ペアの各画像に関連する時間メタデータから決定される、手段と、

所定の時間を示す情報とカレンダーの情報の少なくとも一つに基づいて、前記決定された時間間隔の一つ以上に対する調整値を決定する手段と、

前記複数のイベントクラスタを生成するために前記画像セットにおける前記画像を分割する手段であって、各イベントクラスタに対する分割の境界は、少なくとも前記決定された調整値を用いて決定される、手段と、  
を有することを特徴とする装置。

40

【請求項 14】

複数のイベントクラスタであって、各イベントクラスタが画像セットからの少なくとも一つの画像を含む、複数のイベントクラスタを生成するために前記画像セットを分割するためのシステムであって、

前記システムは、

データとコンピュータプログラムを格納するためのメモリと、

前記コンピュータプログラムを実行するために前記メモリに接続されたプロセッサを

50

有し、

前記コンピュータプログラムは、

前記セットの各画像に対応する時間メタデータを有する画像にアクセスするためのコードと、

前記画像セットにおける隣接した画像の各ペア間の時間間隔を決定するためのコードであって、各前記ペアにおける隣接した画像は、時間的に隣接し、各前記ペアに対する前記時間間隔は、前記ペアの各画像に関連する時間メタデータから決定される、コードと、

所定の時間を示す情報とカレンダーの情報の少なくとも一つに基づいて、前記決定された時間間隔の一つ以上に対する調整値を決定するためのコードと、

前記複数のイベントクラスタを生成するために前記画像セットにおける画像を分割するためのコードであって、各イベントクラスタに対する分割の境界は、少なくとも前記決定された調整値を用いて決定される、コード、

のための命令を含む、  
ことを特徴とするシステム。

#### 【請求項 15】

複数のイベントクラスタであって、各イベントクラスタが画像セットからの少なくとも一つの画像を含む、複数のイベントクラスタを生成するために前記画像セットを分割するためプログラムを有する非一時的なコンピュータ読み取り可能な媒体であって、

前記プログラムは、

前記セットの各画像に対応する時間メタデータを有する画像にアクセスするためのコードと、

前記画像セットにおける隣接した画像の各ペア間の時間間隔を決定するためのコードであって、各前記ペアにおける隣接した画像は、時間的に隣接し、各前記ペアに対する前記時間間隔は、前記ペアの各画像に関連する時間メタデータから決定される、コードと、

所定の時間を示す情報とカレンダーの情報の少なくとも一つに基づいて、前記決定された時間間隔の一つ以上に対する調整値を決定するためのコードと、

前記複数のイベントクラスタを生成するために前記画像セットにおける画像を分割するためのコードであって、各イベントクラスタに対する分割の境界は、少なくとも前記決定された調整値を用いて決定される、コード、

を含む、  
ことを特徴とする非一時的なコンピュータ読み取り可能な媒体。

#### 【発明の詳細な説明】

#### 【技術分野】

#### 【0001】

本発明は、デジタル写真に関し、特に、複数のイベントクラスタを生成するために画像セットを分割するための方法、システム、および装置に関する。本発明はまた、複数のイベントクラスタを生成するために画像セットを分割するためのコンピュータプログラムを記録したコンピュータ読み取り可能な媒体を含むコンピュータプログラム製品に関する。

#### 【背景技術】

#### 【0002】

画像のセットをイベントに分割する(segment)することは、画像により容易にアクセスすることが可能となるように画像を組織化するのに役立つ。イベントはまた、フォトブックにおいて画像を構成するのに役立つ。例えば、イベントからの画像は、フォトブックにおける単一の広がり限定される。イベントは、画像タイムスタンプ、地理位置、および画素情報を含む情報に基づいて決定され得る。

#### 【発明の概要】

#### 【0003】

本発明の目的は、現存の構成の1つ以上の欠点を解決、または少なくとも改善することである。

10

20

30

40

50

## 【 0 0 0 4 】

開示される構成は、入力としての画像タイムスタンプを利用し、かつ、画像間の時間ギャップを調整することにより、画像のセットに対するイベントを決定する便利な方法を伴って課題に対処しようとするものである。時間ギャップは縮小され、当該ギャップは予想市民時間と一致する。本開示のある観点では、時間順の画像のリストにおいて隣接した画像間の時間差が決定され、距離の外れ値 (outliers) がイベント間の境界となるように識別される。予想睡眠時間と一致するギャップの縮小により、単一の日の中で起きるイベントが認識されない可能性に影響を及ぼすことなく、夜間の画像の欠落により分割されにくい、複数日のイベントがもたらされる。

## 【 0 0 0 5 】

本開示のある観点では、週末または複数日の祝祭の睡眠時間と一致するギャップの期間がさらに縮小され、これにより、週末と複数日の祝祭のイベントがより良く検出される。そのようなイベントの例としては、週末の旅行や祝祭の休日である。

## 【 0 0 0 6 】

本開示の別の観点では、複数のイベントクラスタであって、各イベントクラスタが画像セットからの少なくとも一つの画像を含む、複数のイベントクラスタを生成するために前記画像セットを分割する、コンピュータに実装された方法が提供される。

当該方法は、

前記セットの各画像に対応する時間メタデータを有する画像にアクセスする工程と、

前記画像セットにおける隣接した画像の各ペア間の時間間隔を決定する工程であって、各前記ペアにおける隣接した画像は、時間的に隣接し、各前記ペアに対する前記時間間隔は、前記ペアの各画像に関連する時間メタデータから決定される、工程と、

所定の時間を示す情報とカレンダーの情報の少なくとも一つに基づいて、前記決定された時間間隔の一つ以上に対する調整値を決定する工程と、

前記複数のイベントクラスタを生成するために前記画像セットにおける前記画像を分割する工程であって、各イベントクラスタに対する分割の境界は、少なくとも前記決定された調整値を用いて決定される、工程と、

を有する。

## 【 0 0 0 7 】

本開示の別の観点では、複数のイベントクラスタであって、各イベントクラスタが画像のセットからの少なくとも一つの画像を含む、複数のイベントクラスタを生成するために前記画像セットを分割するための装置が提供される。

当該装置は、

前記セットの各画像に対応する時間メタデータを有する画像にアクセスする手段と、

前記画像セットにおける隣接した画像の各ペア間の時間間隔を決定する手段であって、各前記ペアにおける隣接した画像は、時間的に隣接し、各前記ペアに対する前記時間間隔は、前記ペアの各画像に関連する時間メタデータから決定される、手段と、

所定の時間を示す情報とカレンダーの情報の少なくとも一つに基づいて、前記決定された時間間隔の一つ以上に対する調整値を決定する手段と、

前記複数のイベントクラスタを生成するために前記画像セットにおける前記画像を分割する手段であって、各イベントクラスタに対する分割の境界は、少なくとも前記決定された調整値を用いて決定される、手段と、

を有する。

## 【 0 0 0 8 】

本開示の別の観点では、複数のイベントクラスタであって、各イベントクラスタが画像のセットからの少なくとも一つの画像を含む、複数のイベントクラスタを生成するために前記画像セットを分割するためのシステムが提供される。

当該システムは、

データとコンピュータプログラムを格納するためのメモリと、

前記コンピュータプログラムを実行するために前記メモリに接続されたプロセッサを

10

20

30

40

50

有し、前記コンピュータプログラムは、

前記セットの各画像に対応する時間メタデータを有する画像にアクセスするためのコードと、

前記画像セットにおける隣接した画像の各ペア間の時間間隔を決定するためのコードであって、各前記ペアにおける隣接した画像は、時間的に隣接し、各前記ペアに対する前記時間間隔は、前記ペアの各画像に関連する時間メタデータから決定される、コードと、

所定の時間を示す情報とカレンダーの情報の少なくとも一つに基づいて、前記決定された時間間隔の一つ以上に対する調整値を決定するためのコードと、

前記複数のイベントクラスタを生成するために前記画像セットにおける画像を分割するためのコードであって、各イベントクラスタに対する分割の境界は、少なくとも前記決定された調整値を用いて決定される、コード、  
のための命令を含む。

【0009】

本開示の別の観点では、複数のイベントクラスタであって、各イベントクラスタが画像のセットからの少なくとも一つの画像を含む、複数のイベントクラスタを生成するために前記画像セットを分割するためプログラムを有する非一時的なコンピュータ読み取り可能な媒体が提供される。当該プログラムは、

前記セットの各画像に対応する時間メタデータを有する画像にアクセスするためのコードと、

前記画像セットにおける隣接した画像の各ペア間の時間間隔を決定するためのコードであって、各前記ペアにおける隣接した画像は、時間的に隣接し、各前記ペアに対する前記時間間隔は、前記ペアの各画像に関連する時間メタデータから決定される、コードと、

所定の時間を示す情報とカレンダーの情報の少なくとも一つに基づいて、前記決定された時間間隔の一つ以上に対する調整値を決定するためのコードと、

前記複数のイベントクラスタを生成するために前記画像セットにおける画像を分割するためのコードであって、各イベントクラスタに対する分割の境界は、少なくとも前記決定された調整値を用いて決定される、コード、  
を含む。

【0010】

他の観点もまた開示される。

【図面の簡単な説明】

【0011】

本発明の一つ以上の実施形態は、以下の図面を参照して説明される。

【図1】図1は、土曜日に発生する2つのイベントに関連する撮像パターンの例を示す。

【図2】図2は、週末の旅行に関連する撮像パターンの例を示す。

【図3】図3は、複数のイベントクラスタを生成するために画像セットを分割する方法を示す概略的なフロー図である。

【図4】図4は、図3の方法において使用されるタイムスタンプにより順序付けされた画像のシーケンスを示す。

【図5】図5は、図4の画像のシーケンスと、連続する画像間の時間ギャップを示す。

【図6】図6は、連続する画像間の時間差のシーケンスである時間ギャップシーケンスを示す。

【図7】図7は、図3の方法において使用される時間ギャップを調整する方法を示す概略的なフロー図である。

【図8】図8は、図7の方法に従ってギャップを調整する例を示す。

【図9】図9は、図3の方法に従ってギャップ閾値に基づいて画像をイベントにクラスタ化する例を示す。

【図10】図10は、図3の方法において使用されるイベントクラスタを作成する方法を示す概略的なフロー図である。

【図 1 1 A】図 1 1 A は、説明される構成を実行可能な汎用コンピュータシステムの概略的なブロック図を形成する。

【図 1 1 B】図 1 1 B は、説明される構成を実行可能な汎用コンピュータシステムの概略的なブロック図を形成する。

【発明を実施するための形態】

【 0 0 1 2 】

デジタル写真の広がりに伴い、人々が撮像する画像の数は、著しく増加している。しかしながら、画像の数が増加するに従い、画像のコレクションは、管理し、ソートし、画像を見つけることがより難しいものとなっている。

【 0 0 1 3 】

例えば、イベント分割が、フォトアルバム作成のプロセスにおいて早い時点での工程の場合、画像を撮像した時間を示すタイムスタンプは、容易に取得され、特に重要な処理をほとんど必要としない。画像のタイムスタンプのみに基づいて画像のセットに対するイベントを決定する一つの方法は、時間順の画像のリストにおいて隣接した画像間の時間差を決定し、距離の外れ値を識別する。外れ値は、全てのイベントの境界に対する候補を表す。このような方法は効果的であるが、複数日を占めるイベントが見当たらなくなる傾向がある。

【 0 0 1 4 】

イベントを識別するためにタイムスタンプのみを用いてイベントを決定する従来の方法は、画像はイベントにおいてより多く撮像されるという仮定に基づいている。このような従来の方法はまた、2つの画像間の相対的に大きいギャップが、イベントが終わること、および/またはイベントが始まることを識別することを仮定する。方法は、週末の旅行のような、複数日を占めるイベントを認識しない可能性がある。なぜならば、撮影者が就寝の間の撮像におけるギャップは、一日の間で発生する同じギャップと同じウェイトを有するからである。

【 0 0 1 5 】

画像のコレクションを整理し、画像コレクションの異なる部分をユーザが認識する手助けをするために、画像の一群をイベントにグループ化する方法を以下に示す。説明する方法は、フォトアルバム生成アプリケーションにおいても使用されてもよく、ここで、画像のコレクションは、イベントに分割され、どのイベントをフォトアルバムにおいて表すかをユーザに選択させることを許容する。

【 0 0 1 6 】

上述したように、画像のタイムスタンプは容易に取得され、処理をほとんど必要とせず、これは、イベント分割が、フォトアルバム作成のプロセスにおける早い時点での工程の場合に有利である。説明する方法の一つの構成において、画像のセットの画像のタイムスタンプに対応する時間ギャップのシーケンスが決定され、当該時間ギャップのシーケンスは、例えば、撮影者が睡眠している (sleeping) と見込まれるときの時間間隔といった、撮影者が撮像不可であることを考慮して修正される。修正された時間ギャップのシーケンスは、その後、画像をイベントにクラスタ化するために使用され得る。

【 0 0 1 7 】

図11Aと図11Bは、説明する多様な構成を実行可能な汎用コンピュータシステム1100を示す。

【 0 0 1 8 】

図11Aでわかるように、コンピュータシステム1100は、コンピュータモジュール1101、キーボード等の入力デバイス1102、マウスポインタデバイス1103、スキャナ1126、カメラ1127、マイクロホン1180、プリンタを含む出力デバイス1115、ディスプレイデバイス1114、および、ラウドスピーカ1117を含む。外部の変調復調器 (モデム (Modem)) 送受信デバイス1116は、接続1121を介して、通信ネットワーク1120へ、または通信ネットワーク1120から通信を行うために使用され得る。通信ネットワーク1120は、ワイドエリアネットワーク (WAN) であり、例えば、インターネット、セルラ電気通信ネットワーク、またはブ

10

20

30

40

50

ライベートWANである。接続1121は、電話線であり、モデム1116は、従来の「ダイヤルアップ」モデムであり得る。代わりに、接続1121は、高容量（例えばケーブル）接続であり、モデム1116は、ブロードバンドモデムであり得る。無線モデムもまた、通信ネットワーク1120への無線接続のために使用され得る。クラウドベースの構成のために、一つ以上のサーバ1190がネットワーク1120に接続され得る。

#### 【0019】

コンピュータモジュール1101は、典型的には、少なくとも一つのプロセッサユニット1105とメモリユニット1106を含む。例えば、メモリユニット1106は、半導体ランダムアクセスメモリ（RAM）と半導体読み出し専用メモリ（ROM）を有し得る。コンピュータモジュール1101はまた、ビデオディスプレイ1114、ラウドスピーカ1117、およびマイクロホン1180に接続されたオーディオ/ビデオインタフェース1107、キーボード1102、マウス1103、スキャナ1126、カメラ1127、およびオプションとしてジョイスティックや他のヒューマンインタフェースデバイス（不図示）に接続されたI/Oインタフェース1113、および、外部モデム1116とプリンタ1115のためのインタフェース1108を含む。実装によっては、モデム1116は、例えばインタフェース1108内といった、コンピュータモジュール1101内に組み込まれる。コンピュータモジュール1101はまた、ローカルエリアネットワーク（LAN）として知られるローカルエリア通信ネットワーク1122との接続1123を介してコンピュータシステム1100との接続を許容する、ローカルネットワークインタフェース1111を有する。図11Aに図示されるように、ローカル通信ネットワーク1122はまた、典型的にはいわゆる「ファイヤー・ウォール」デバイスや同様の機能を有するデバイスを有する接続1124を介してワイドネットワーク1120に接続し得る。ローカルネットワークインタフェース1111は、イーサネット回路カード、Bluetooth（登録商標）無線工程、またはIEEE 802.11無線工程を含み得る。しかしながら、多数の他のタイプのインタフェースは、インタフェース1111に対して実施され得る。

#### 【0020】

I/Oインタフェース1108と1113は、シリアル接続およびパラレル接続の両方を行うことができ、前者は、典型的には、ユニバーサル・シリアル・バス（USB）に従って実装され、対応するUSBコネクタ（不図示）を有する。記憶装置1109は、典型的には、ハードディスクドライブ（HDD）1110を含んで提供される。フロッピーディスクや磁気テープ装置（不図示）などの他の記憶装置も使用され得る。光ディスクドライブ1112は、データの揮発性のソースとして動作するために提供される。光ディスク（例えば、CD-ROM、DVD、ブルーレイ（登録商標）ディスク）、USB-RAM、ポータブルまたは外部ハードディスク、フロッピーディスクなどのポータブルメモリデバイスは、例えば、システム1100へのデータの適切なソースとして使用され得る。

#### 【0021】

コンピュータモジュール1101の構成要素1105～1113は、典型的には、相互接続バス1104を介して、関連する分野の当業者に知られているコンピュータモジュール1100の動作の従来のモードとなるような方法で通信する。例えば、プロセッサ1105は、接続1118を用いてシステムバス1104と接続する。同様に、メモリ1106と光ディスクドライブ1112は、接続1119によりシステムバス1104に接続される。説明する構成が実施されるコンピュータの例として、IBM-PCの、および互換性のある、Sun Sparcstation、Apple MAC（登録商標）、または同様のコンピュータシステムを含む。

#### 【0022】

説明する方法は、コンピュータシステム1100を用いて実装され得る。ここで、説明する図1～図10のプロセスが、コンピュータシステム1100内で実行可能な一つ以上のソフトウェアアプリケーションプログラム1133として実装され得る。特に、説明する方法の工程は、コンピュータシステム1100内で実行されるソフトウェア1133における命令1131（図11Aを参照）により達成される。ソフトウェアの命令1131は、それぞれが一つ以上の特定のタスクを実行するための、一つ以上のコードモジュールとして形成され得る。ソフトウェアはまた、2つのパートに分割され得る。第1のパートと対応するコードモジュールは説明す

10

20

30

40

50

る方法を実行し、第2のパートと関連するコードモジュールは、第1のパートとユーザ間のインタフェースを管理する。

【0023】

ソフトウェアは、例えば、以下に説明する記憶装置を含むコンピュータ読み取り可能な媒体に格納される。ソフトウェア1133は、典型的には、HDD1110またはメモリ1106に格納される。ソフトウェアは、コンピュータ読み取り可能な媒体から、コンピュータシステム1100にロードされ、コンピュータシステム1100により実行される。従って、例えば、ソフトウェアは、光ディスクドライブ1112により読み取りされる、光学的に読み取り可能なディスク記憶媒体（例えばCD-ROM）1125に格納され得る。コンピュータ読み取り可能な媒体に記録された、そのようなソフトウェアまたはコンピュータプログラムを有するコンピュータ読み取り可能な媒体は、コンピュータプログラム製品である。コンピュータシステム1100におけるコンピュータプログラム製品の使用により、説明する方法を実装する有利な装置が効果的にもたらされる。

【0024】

いくつかの例において、アプリケーションプログラム1133は、一つ以上のCD-ROM上で符号化されて、ユーザに提供され、対応するドライブ1112を介して読み取られ得る。または、代わりに、ネットワーク1120または1122から、ユーザにより読み取られ得る。さらに、ソフトウェアは、他のコンピュータ読み取り可能なメディアから、コンピュータシステム1100へロードされ得る。コンピュータ読み取り可能な記憶媒体は、実行および/または処理のために、記録された命令および/またはデータを、コンピュータシステム1100に提供する、あらゆる非一時的な有形の記憶媒体を参照する。そのような記憶媒体の例として、コンピュータモジュール1101の内部または外部であるかを問わず、フロッピーディスク、磁気テープ、CD-ROM、DVD、ブルーレイ（登録商標）ディスク、ハードディスクドライブ、ROMまたは集積回路、USBメモリ、光磁気ディスク、またはPCMCIA等のコンピュータ読み取り可能なカード等を含む。コンピュータモジュール1101への、ソフトウェア、アプリケーションプログラム、命令、および/またはデータの提供に関与し得る、一時的または非有形のコンピュータ読み取り可能な送信媒体は、無線または赤外線送信チャネル、並びに、別のコンピュータまたはネットワーク化されたデバイスへのネットワーク接続、および、e-mail送信およびウェブサイト等に記録された情報を含むインターネットまたはイントラネット等を含む。

【0025】

上述した、アプリケーションプログラム1133の第2のパートと対応するコードモジュールは、表示される、そうでなければ、ディスプレイ1114上に表される、一つ以上のグラフィックユーザインタフェース（GUI）を実装するために実行され得る。典型的にはキーボード1102およびマウス1103の操作を介して、コンピュータシステム1100とアプリケーションのユーザは、操作コマンドおよび/またはGUIに関連するアプリケーションへの入力を提供するために、機能的に調整可能な方法で、インタフェースを操作し得る。ラウドスピーカ1117を介したスピーチプロンプトの出力およびマイクロフォン1180を介した音声コマンドの入力を利用するオーディオインタフェースのような、機能的に調整可能なユーザインタフェースの他の形態もまた、実装され得る。

【0026】

図11Bは、プロセッサ1105と「メモリ」1134の詳細なブロック図である。メモリ1134は、図11Aにおけるコンピュータモジュール1101によりアクセス可能な全てのメモリモジュール（HDD1100と半導体メモリ1107を含む）の論理的なアグリゲーションを表す。

【0027】

コンピュータモジュール1101が最初に電源オンされると、power-on self-test（POST）プログラム1150が実行する。POSTプログラム1150は、典型的には、図11Aにおける半導体メモリ1106のROM1149に格納される。ソフトウェアを格納するROM1149のようなハードウェアデバイスは、時によりファームウェアと参照される。POSTプログラム1150は、コンピュータモジュール1101内で、適切な機能を保証するために、ハードウェアを検査し、典型的



には、プロセッサ1105、メモリ1134 ( 1109、1106 )、典型的にはROM1149に格納されているbasic input-output system software ( BIOS ) モジュール1151を、正しい動作のためにチェックする。一度POSTプログラム1150が動作に成功すると、BIOS1151は、図11Aのハードディスクドライブ1110を起動する。ハードディスクドライブ1110の起動により、ハードディスクドライブ1110上に記録されるブートストラップ・ローダーが、プロセッサ1105を介して実行される。これは、オペレーティングシステム1153をRAMメモリ1106へロードし、そこで、オペレーティングシステム1153は動作を開始する。オペレーティングシステム1153は、プロセッサ管理、メモリ管理、デバイス管理、ストレージ管理、ソフトウェアアプリケーションインタフェース、一般ユーザのインタフェースを含む多様なハイレベルな機能を果たすための、プロセッサ1105により実行可能なシステムレベルアプリケーションである。

10

#### 【 0 0 2 8 】

オペレーティングシステム1153は、コンピュータモジュール上で動作する各プロセスまたはアプリケーションが、別の目的のために割り当てられたメモリと衝突せずに実行する好適なメモリを有することを保証するために、メモリ1134 ( 1109、1106 ) を管理する。さらに、図11Aのシステム1100における異なるタイプのメモリが、各プロセスが効率的に動作できるように、適切に使用される必要がある。従って、集約メモリ1134は、メモリの特定の部分をどのように割り当てるかを示すことは意図されず、コンピュータシステムによりアクセス可能なメモリと、どのようにそのようなものが使用されるかの一般的な概念を提供するためのものである。

20

#### 【 0 0 2 9 】

図11Bに示すように、プロセッサ1105は、制御部1139、演算論理装置 ( ALU ) 1140、および、時にキャッシュメモリと呼ばれる局所的または内部のメモリ1148を含む、複数の機能モジュールを有する。キャッシュメモリ1148は、典型的には、レジスタ部において、複数の記憶レジスタ1144 ~ 1146を有する。一つ以上の内部バス1141は、機能的にこれらの機能モジュールと内部接続する。プロセッサ1105は、典型的には、接続1118を用いて、システムバス1104を介して、外部の機器と通信するために一つ以上のインタフェース1142も有する。メモリ1134は、接続1119を用いてバス1104と接続される。

#### 【 0 0 3 0 】

アプリケーションプログラム1133は、条件付き分岐とループの命令を含み得る一連の命令1131を含む。プログラム1133はまた、プログラム1133の実行において使用されるデータ1132を含み得る。命令1131とデータ1132は、それぞれ、メモリ位置1128、1129、1130と、1135、1136、1137に格納される。命令1131の相対的なサイズとメモリ位置1128 ~ 1130に従い、特定のメモリが、メモリ位置1130に示される命令により表される単一のメモリ位置において格納され得る。また、命令は、メモリ位置1128と1129示される命令部分により表されるように、それぞれが分離したメモリ位置に格納されるいくつかのパートに分割され得る。

30

#### 【 0 0 3 1 】

一般的に、プロセッサ1105は、そこで実行される命令のセットが与え得られる。プロセッサ1105は、次の入力を待ち、プロセッサ1105は、命令の別のセットを実行することにより反応する。各入力、一つ以上のソースから提供され得る。当該ソースは、入力装置1102、1103の一つ以上により生成されたデータ、ネットワーク1120、1102のうちの一つを渡る外部ソースから受信したデータ、記憶装置1106、1109のうちの一つから取り出されたデータ、または対応するリーダ1112に挿入された記憶媒体1125から取り出されたデータを含み、これらは全て図11Aに表現されている。命令のセットの実行は、いくつかのケースにおいて、データの出力をもたらす。実行はまた、メモリ1134にデータまたは変数を格納することを含み得る。

40

#### 【 0 0 3 2 】

開示する構成は、入力変数1154を使用し、これは、対応するメモリ位置1155、1156、1157においてメモリ1134に格納される。開示する構成は、出力変数1161を生成し、これは、

50

対応するメモリ位置1162、1163、1164においてメモリ1134に格納される。中間変数1158は、メモリ位置1159、1160、1166、1167において格納され得る。

【0033】

図11Bのプロセッサ1105を参照すると、レジスタ1144、1145、1146、演算論理装置（ALU）1140、および制御部1139は、共に動作し、プログラム1133を構成する命令セットにおける全ての命令に対する「フェッチ、復号、および実行」のサイクルを実行する必要がある一連のマイクロオペレーションを実行する。各フェッチ、復号、および実行のサイクルは、  
フェッチオペレーション（メモリ位置1128、1129、1130から命令1131をフェッチまたは読み出す）。 10

復号オペレーション（制御部1139がどの命令がフェッチされたかを決定する。）

実行オペレーション（制御部1139および／またはALU1140が命令を実行する。）  
を、含む。

【0034】

その後、次の命令に対する更なるフェッチ、復号、実行のサイクルが実行され得る。同様に、制御部1139が値をメモリ位置1132に格納または書き込むことにより、記憶サイクルが実行され得る。

【0035】

図1から10のプロセスにおける各工程またはサブプロセスは、プログラム1133の一つ以上の部分に割り当てられ、プログラム1133の当該言及した部分に対する命令セットにおける全ての命令に対してフェッチ、復号、および実行が、共に実施されるように、プロセッサ1105におけるレジスタ部1144、1145、1147、ALU1140、および制御部1139により実行される。 20

【0036】

説明する方法はまた、説明する方法の機能やサブ機能を実行する一つ以上の集積回路等の、固有のハードウェアにおいて実装され得る。このような固有のハードウェアは、グラフィックプロセッサ、デジタル信号プロセッサ、または一つ以上のマイクロプロセッサ、および付随するメモリを含み得る。

【0037】

図1および図2は、撮像のパターンの2つの例である。図1は、土曜の朝と土曜の夜に起こる2つのイベントに対する撮像パターンを示している。軸120は、2時間間隔を示し、軸110は、特定の2時間間隔に撮像された画像の数を示す。土曜午前8時（8am）から土曜午後12時（12pm）の期間130において、20枚の画像が撮像され、続いて、土曜午後4時から午前12時の間隔150において撮像される画像までの、少なくとも4時間の時間間隔160を表すギャップが続く。 30

【0038】

図2は、週末の旅行イベントに対する撮像パターンを示すグラフ200である。図2からわかるように、金曜夜210、土曜230、240、および日曜260において撮像アクティビティが発生している。金曜午後10時（10pm）から土曜午前6時（6am）までのギャップ220と、土曜午前12時から日曜午前6時までのギャップ250が存在する。ギャップ220は、少なくとも8時間の時間間隔を示し、ギャップ250は、少なくとも6時間の時間間隔を表す。2つのギャップ220と250は、撮影者が就寝中の時間間隔に関連する可能性が高い。ギャップ220と250は、図1の4時間のギャップ160より長い。 40

【0039】

以下に説明する方法は、図1の画像を朝と夜のイベントに分割するものの、図2の単一の週末の旅行イベントを分割しないように使用される。

【0040】

時間ギャップのシーケンスの更なる改良が、週末の旅行や休日の旅行といった、より多くの特定の複数日のイベントの認識を向上させるために、使用される。

【0041】

複数のイベントクラスタを生成するための画像セットを分割する方法300は、図3を参照して以下に説明される。すでに説明したように、各イベントクラスタは、画像セットからの少なくとも一つの画像を含む。方法300は、ハードディスクドライブ1110上に存在するソフトウェアアプリケーションプログラム1133の一つ以上のソフトウェアコードモジュールとして実装され、プロセッサ1105によりその実行が制御され得る。方法300は、図4において見られる例示的な画像セット410（記憶装置1109内に構成されている画像セット410）を参照した例示の方法により説明される。

【0042】

方法300は、アクセスする工程310で開始する。ここで、画像セットにおける画像に対応するタイムスタンプが、プロセッサ1105の実行下でアクセスされる。画像セットにおける各画像は、当該画像に関連付けられたタイムスタンプを有し、タイムスタンプは、当該画像の撮像の時間を示す。タイムスタンプは、画像に対するデジタルファイルに組み込まれたexchangeable image file format（EXIF）データのような、メタデータ属性において記録され得る。そのようなタイムスタンプは、「画像メタデータ」または「時間メタデータ」として参照され得る。別の構成では、画像の撮像の時間は、画像に対するデジタルファイルと別に（例えばデータベースにおいて）記録され得る。さらなる構成では、画像ファイル作成時間が、撮像時間として解釈され得る。

【0043】

タイムスタンプは、ストレージに対する複数の異なるフォーマット、およびソフトウェアシステムによる操作へと変換され得る。Unixタイムスタンプ、一連の日時、またはJava Calendar classなどのソフトウェアクラスを含むタイムスタンプを記憶し、操作するための好適なフォーマットが方法300において使用され得る。2つのタイムスタンプ間の時間差を決定するための好適な方法も、記憶されたタイムスタンプから日時（date and time of day）を決定するために方法300において使用され得る。

【0044】

方法300は、その後工程310から順序付けする工程320へ進み、ここで、画像セットにおける画像は、各画像に関連付けられたタイムスタンプに従って順序付けされる。工程320は結果として「順序付けされたタイムスタンプのリスト」を生成する。順序付けされたタイムスタンプのリストは、プロセッサ1105によるメモリ1106において格納され得る。

【0045】

図4は、例示的な、順序付けされたタイムスタンプのリスト420を示す。図4からわかるように、順序付けされたタイムスタンプのリスト420は、タイムスタンプのリストの要素430、440、450、460、470、480、490を含む。タイムスタンプのリストの要素430、440、450、460、470、480、490のそれぞれは、画像セット410の画像（または「写真」）に関連する。タイムスタンプのリスト420は、タイムスタンプ430、440、450、470、480、490に基づいて順序付けされる。例えば、画像435に対するタイムスタンプのリストの要素430は、画像445に対するタイムスタンプのリストの要素440に対するタイムスタンプ以下の値を有する。すなわち、タイムスタンプのリストの要素430の $t_1$ は、タイムスタンプのリストの要素440に対する $t_2$ 以下となる（ $t_1 \leq t_2$ ）。順序付けされたタイムスタンプのリスト420は、画像のセットのタイムスタンプを表す、多くの可能性のある方法のうちの一つの例である。

【0046】

工程320に続いて、方法300は、決定する工程330へ進み、ここで、隣接（すなわち連続する）画像（例えば画像435と440）の各ペア間の時間間隔を表す時間ギャップが、プロセッサ1105の実行の下、決定される。各ペアにおける隣接する画像は、時間順のシーケンスにおいて（時間的に）隣接する。画像の各ペア間の時間間隔を表す時間ギャップは、ペアにおける各画像に関連づけられる時間メタデータから決定される。

【0047】

工程330で決定された時間ギャップは、図5においてみられるように、「時間ギャップリスト」510において格納され、例えばメモリ1106内に構成される。図5を参照すると、時間

10

20

30

40

50

ギャップのリスト510は、順序付けされたタイムスタンプのリスト420から決定される。図5の例において、時間ギャップのリストの要素530は、 $\text{ギャップ} = t_2 - t_1$ を有する。ここで、 $t_1$ は、画像435に対するタイムスタンプのリストの要素430のタイムスタンプであり、 $t_2$ は、画像445に対するタイムスタンプのリストの要素440に対するタイムスタンプである。

【 0 0 4 8 】

図6は、51の画像を有する画像セットに対する時間ギャップのリスト600の図式の例を示す。軸620は、ギャップの番号であり、軸610は、時間ギャップ（分）である。例えば、ギャップ番号10（630）は、画像（10）と（11）との間の時間間隔を表すギャップに対応し、画像（10）と（11）の撮像の間で、600分の間隔があったことを示している。時間ギャップのリスト600は、連続する画像間の時間ギャップを表す、多くの可能性のある方法のうちの一つの例である。

【 0 0 4 9 】

方法300はその後、工程330から調整する工程340へ進み、ここで、工程330で決定された時間ギャップは、プロセッサ1105の実行の下、調整される。以下に説明するように、時間ギャップは、日の所定の時間の情報およびカレンダーの情報のうちの少なくとも一つに基づいて、一つ以上の時間ギャップ（または時間間隔）に対して決定された調整値に基づいて調整される。工程340での時間ギャップの調整は、時間ギャップのリストおよび順序付けされたタイムスタンプのリストを活用することにより、時間ギャップのリストを修正する。工程340で実行される、時間ギャップを調整する方法700は、図7を参照して以下でさらに説明する。

【 0 0 5 0 】

方法300はその後、作成する工程350へ進み、ここで、調整された時間ギャップを用いてイベントクラスタが作成される。工程350で実行される、イベントクラスタを作成する方法1000は、図10を参照して以下で詳細に説明する。方法1000に従って、イベントクラスタは、ギャップにより示される時間間隔とクラスタの時間の時間範囲との比較により作成される。以下に詳細に説明するように、方法1000は、一つ以上のイベントクラスタを生成するために、画像セット（例えば画像セット420）における画像を分割するように構成される。一つの構成において、複数のイベントクラスタが決定され得る。各イベントクラスタに対する分割の境界は、工程340で決定される調整値を用いて決定され得る。

【 0 0 5 1 】

工程350で実行され得る、イベントクラスタを作成する別の方法900もまた、以下に説明する。

【 0 0 5 2 】

工程340で実行される、時間ギャップを調整する方法700を、図7を参照して説明する。方法700は、ハードディスクドライブ1110において存在するソフトウェアアプリケーションプログラム1133の一つ以上のソフトウェアコードモジュールとして実装され、プロセッサ1105による実行において制御され得る。図7においてわかるように、方法700では、ループが実装され、そこで、工程730、740、750、および760を実装するためにソフトウェア機能が呼ばれる。他の構成において、時間ギャップは、「非睡眠（non-sleeping）」期間において時間ギャップを増加させることにより調整され得る。

【 0 0 5 3 】

方法700を、図4の例示的なタイムスタンプのリスト420を参照した例の手法により説明する。方法700は、アクセスする工程710で開始し、ここでタイムスタンプのリスト420のタイムスタンプのリストの要素430が、プロセッサ1105の実行の下、アクセスされる（例えばハードディスクドライブ1110から）。図4からわかるように、最初のタイムスタンプのリストの要素430は、画像セット410における最初の画像435に関連する。

【 0 0 5 4 】

方法700に従って、工程730、740、750および760が、順序付けされたタイムスタンプのリスト420の最後のタイムスタンプのリストまで繰り返される。したがって、決定する工

10

20

30

40

50

程720では、タイムスタンプの要素が順序付けされたタイムスタンプのリスト420の最後の処理されるタイムスタンプの要素（すなわち、画像セット410の最後の画像に対応する）である場合に、方法780は終了する。それ以外の場合は、方法700は、決定する工程730へ進む。

【0055】

方法700の最初のループでは、決定する工程730で、タイムスタンプの要素430のタイムスタンプに対応する日時（date and time of day）が決定される。上述したように、最初のタイムスタンプのリストの要素430は、画像セット410における最初の画像435に関連する。この例において、画像435は、「現在の画像」（すなわち現在処理される画像）として参照される。従って、決定する工程730で決定される日時は、現在の画像が撮像された日時である。いくつかの構成では、タイムスタンプは、工程730で処理の必要がない場合に、日時のフォーマットで記憶される。それ以外は、他のフォーマットで記憶される場合、タイムスタンプからの日時を決定する好適な方法が工程830で使用され得る。

【0056】

その後、選択する工程740では、タイムスタンプの要素430と次のタイムスタンプの要素440に対応する時間ギャップのリスト510におけるギャップ530が、プロセッサ1105の実行下で選択される。図4の例では、タイムスタンプの要素430は、画像430（すなわち現在の画像）に対応し、タイムスタンプの要素440は、画像445（すなわち次の画像）に対応する。

【0057】

方法700は、次の決定する工程750へ続き、ここで、「睡眠（sleeping）」期間と重なる、選択されたギャップ530における割合が、プロセッサ1105の実行下で決定される。工程750で決定された割合は、調整値として参照され得る。調整値は、選択されたギャップ530と睡眠の期間（すなわち、1日周期の期間（time of day period））の割合により表される時間間隔の重なりに基づいて決定される。その後、工程760で、選択されたギャップ530は、工程750で決定された調整値の割合（すなわち、調整値）と等しい量だけ縮小（減少）される。

【0058】

方法700は、工程760から選択する工程770へ進み、ここで、画像445（すなわち次の画像）に対応する順序付けされたタイムスタンプのリスト420における次のタイムスタンプの要素440が、プロセッサ1105の実行の下、選択される。工程770に続いて、方法700は工程720へ戻る。

【0059】

工程750と760で行われるギャップの調整を、図8を参照した例の手法により説明する。上述したように、ギャップが「予想睡眠時間」に対応した、1日周期の期間（time of day period）と重なる場合に、ギャップを縮小するために、工程750と760でギャップの調整が行われる。図8の例では、予想睡眠時間はあらかじめ定義され、ある日の午後11時（11pm）から次の日の午前7時（7am）の間になるように設定される。図8における、予想睡眠時間は、固定であり、日々繰り返され、一般の人々の予想睡眠時間に対応する。他の睡眠時間も使用され得る。例えば、予想睡眠時間は、国ごとに変化し得る。睡眠時間の期間もまた、夏の月間と冬の月間で異なる睡眠パターンを有する人々と共に変化し得る。これは、所与の画像セット内では、異なる予想睡眠時間が適用され得ることを意味する。

【0060】

他の構成では、工程350のように睡眠時間と重なるギャップを縮小するのではなく、睡眠時間と重ならないギャップを拡大（増大）してもよい。調整値は、時間間隔と睡眠時間（すなわち、時間間隔）の重ならない部分に基づいて決定され得る。睡眠時間と重ならないギャップを拡大することは、睡眠時間と重なるギャップを縮小することと同様の効果を有する。よって、工程350のように、睡眠時間と重ならないギャップを縮小させる構成のみを以下に説明する。

【0061】

図8の例において、2つの画像803と808が、同じ日の間の日中近くで撮像される。ギャップ806は予想睡眠時間と重ならず、ギャップ805に対する調整はない。タイムスタンプ $t_k$ を有する画像「k」813と、タイムスタンプ $t_{k+1}$ を有する画像「k+1」820の2つの画像813が、予想睡眠時間がギャップ $815 = t_{k+1} - t_k$ 内となるように、連続する日において撮像される。図8の例において、予想睡眠時間の期間である8時間よりギャップ815は大きい（ギャップ $815 > 8$ 時間）。図8の例では、ギャップ815は、以下の式（1）に従って、予想睡眠時間の期間の割合により縮小される。

$$\text{新しいギャップ} = \text{古いギャップ} - ( \quad * o ) \quad (1)$$

ここで、 $0 < \quad 1$ であり、 $o$ は古いギャップと予想睡眠時間の重なり期間である。

【0062】

ギャップを調整するための調整値を決定する他の構成も使用され得る。一つの他の構成において、ギャップは、別の関数を用いて、重なりに適用された単純な割合に縮小され得る。例えば、工程750と760でのギャップの縮小を決定するために、対数関数が使用され得る。

【0063】

画像「j+1」829と画像「j+2」は、予想睡眠時間内で撮像される。式（1）に従い、ギャップ826は $(t_{j+1} - t_j) - ( \quad * o1 )$ となるように調整される。ここで $o1$  850は、予想睡眠時間とのギャップ826の重なりである。同様に、調整された場合のギャップ832は $(t_{j+2} - t_{j+1}) - ( \quad * ((t_{j+2} - t_{j+1})) )$ になり、調整された場合のギャップ841は、 $(t_{j+3} - t_{j+2}) - ( \quad * o2 )$ になり、ここで $o2$  は、予想睡眠時間とのギャップ841の重なりである。

【0064】

従って、画像が予想睡眠時間内で撮像される場合に、ギャップの調整はさらに適用することが可能である。通常と異なる時間で撮像された画像は、典型的には、特異な目的を示す。これは、画像は、同じイベントのものと考慮されるべきであり、画像は、ギャップが縮小されなかった場合に起こり得る異なるイベントに分割されるべきではないことを意味する。しかしながら、予想睡眠時間における持続的な撮像は、予想睡眠時間が撮影者の睡眠パターンと合致せず、別の構成が使用され得ることを示し得る。例えば、一つの別の構成において、最も大きい2つのギャップ826と841のみが式（1）に従って調整される。

【0065】

イベントクラスタを作成する工程350を、図9と図10を参照した例の手法により、ここでさらに説明する。

【0066】

図9の例において、調整された時間ギャップを用いてイベントクラスタを作成するために、ギャップの閾値が使用される。時間ギャップのリスト900は、51の画像のセットに対応した50のギャップを有する。軸920は、ギャップの番号を表し、軸910は、ギャップ時間（分）を表している。図9の例では、430分の値の閾値015が使用される。430分以上のギャップが、イベント間の切れ目と考えられる。

【0067】

図9の例では、それぞれが430分以上の時間間隔を表す3つのギャップ930、940、960が存在する。ギャップ（10）930は、画像（10）と（11）間のギャップに対応するので、画像（1）から（10）は、包括的にイベント（1）を形成する。画像（1）から（10）は、51の画像のセットの第1のセグメントを表す。また、画像（11）から（20）は、包括的にイベント（2）を形成し、画像（11）から（20）は、51の画像のセットの第2のセグメントを表す。さらに、画像（21）から（34）は、包括的にイベント（3）を形成し、51の画像のセットの第3のセグメントを表す。残りの画像である画像（43）から（51）は、イベント（4）を形成し、51の画像のセットの第4のセグメントを表す。各イベントクラスタ間の境界（またはセグメント）は、ギャップの閾値を用いて決定される。

【0068】

工程350で実行される、イベントクラスタを作成する方法1000を、図10を参照して説明する。方法1000は、調整された時間ギャップを用いたイベントクラスタリングの別の構成

10

20

30

40

50

である。方法1000において、ギャップの閾値は、固定ではないが、ある範囲内であり、それまでのクラスタの時間期間と共に増加する。方法1000は、調整されたギャップにより表される調整時間間隔と、一様な (even) クラスタに関連する時間間隔との比較に基づいて、画像をイベントクラスタに割り当てるように構成される。

【0069】

方法1000は、複数日に渡る持続的な画像の撮像が、病気やカメラのバッテリーがチャージされていないなどの理由で画像が撮像されなかった日により分断され得る休暇に対して、有利である。休暇は、長い時間ギャップに起因した2つのイベントに分断されず、方法1000は一つのイベントを作成しやすいことが望ましい。

【0070】

方法1000は、ハードディスクドライブ1110に存在するソフトウェアアプリケーションプログラム1133の一つ以上のソフトウェアコードとして実装され、プロセッサ1105による実行において制御され得る。方法1000は、順序付けされた時間リスト420を参照した例の手法により説明する。

【0071】

方法1000は、作成する工程1010から開始し、ここで最初のクラスタが作成され、プロセッサ1105によりメモリ1106へ記憶される。その後、選択する工程1020では、画像セット410の最初の画像435が現在の画像として選択される。画像435は、順序付けされたタイムスタンプのリスト420から、タイムスタンプのリストの要素430を選択することにより選択される。

【0072】

工程1030、1040、1050、1060、1070、および1080は、順序付けされたタイムスタンプのリスト420において連続する画像に対して実行される。図10においてわかるように、方法1000においてはループが実装され、そこで、工程1030、1040、1050、1060、1070、および1080を実装するためにソフトウェア機能が呼ばれる。

【0073】

最初のループでは、割り当てる工程1030で、現在の画像 (すなわち画像435) が現在のクラスタに割り当てられる。

【0074】

その後、決定する工程1040で、プロセッサ1105が、順序付けされたタイムスタンプのリスト420における最後の画像に対するチェックのために使用され、満たされた場合に方法1000は完了する。最後の画像は、最後のタイムスタンプのリストの要素 (例えばタイムスタンプのリストの要素490) に対応する。それ以外は、調整されたギャップにより表される、次の画像445までの時間間隔が、取得する工程1050で取得される。

【0075】

その後、決定する工程1060では、次の画像445までの調整されたギャップが「クラスタ時間の長さ」の関数の値以上であれば、方法1000は作成する工程1070へ進む。クラスタ時間の長さは、現在のクラスタにおける最初の画像435と現在のクラスタにおける最後の画像を表す現在の画像との間の時間である。一つの構成では、クラスタ時間の長さは、順序付けされたタイムスタンプのリスト420における連続した画像間の調整された時間ギャップにより表される時間間隔の累積和と同じである。別の構成では、クラスタ時間の長さは、クラスタにおける最初の画像と現在の画像との間のタイムスタンプの差と等しい。クラスタにおける最初の画像と現在の画像との間のタイムスタンプの差は、調整されていない時間ギャップの和と同等である。

【0076】

工程1060で実施されるテストは、以下の式 (2) に従って表され得る：

次の画像までの調整された時間ギャップ (クラスタ時間の長さ) (2)

ここで、 は、一つの構成であり、式 (3) により定義される単調増加関数である：

$$= \quad + \quad * \log (1 + (\text{クラスタ時間の長さ}) / \quad ) \quad (3)$$

パラメータ 、 、 および の定数は固定である。

10

20

30

40

50

## 【 0 0 7 7 】

クラスタ時間の長さの関数 の他の構成は、指数関数、多項式の関数、およびステップ関数を含むことが可能である。

## 【 0 0 7 8 】

作成する工程1070で、新しいイベントクラスタが作成される。その後、次の画像の工程1080で、順序付けされたタイムスタンプのリスト420におけるタイムスタンプのリストの要素440を選択することにより、方法は次の画像445へ進む。次の画像445は、方法1000の目的に対して現在の画像となる。方法1000は、工程1030へ戻り、工程1070へ続く。

## 【 0 0 7 9 】

別の構成では、時間ギャップの調整は、イベントクラスタリングに影響を与えるためにも使用され、ここで週末における予想睡眠時間、複数日の祝祭、および、認められた休日と一致するギャップの期間はさらに縮小され、週末の旅行、祝祭の旅行、および休暇の良好な検出がもたらされる。

10

## 【 0 0 8 0 】

一つの構成において、方法300における全ての工程310、320、330、および350は、工程340でのギャップの調整を除いて上述のように実行される。そのような構成において、工程750では、週末、複数日の祝祭および認められた休日の期間の間に起こる予想睡眠時間が、工程760でさらにギャップを縮小させるために使用される。例えば、予想睡眠時間は、ある日の午後11時（11pm）から次の日の午前9時（9am）になるように設定され、次の日は認められた休日である。ギャップにより表される時間間隔は、重なる期間が週末、休暇のシーズンまたは複数日の祝祭にかかる場合は、さらに縮小（減少）され得る。

20

## 【 0 0 8 1 】

そのような複数日の祝祭と一般の休日の期間の日付は、年毎に変わり得る。クラウドベースの構成では、複数日の祝祭と一般の休日の期間の日付が保存されたソフトウェアカレンダーは、新しい複数日の祝祭と一般の休日の期間の日付が公になった際に更新されることが可能である。例えば、ソフトウェアカレンダーは、サーバ1190に格納され得る。他の構成では、モバイルベースとしてもよく、説明した方法を実装したソフトウェアプログラム1133は、定期的なソフトウェアアップグレードの方法によりアップデートされ得る。ソフトウェアプログラム1133は、記憶媒体1109内に格納され、ソフトウェアプログラム1133は、新しい日付を追加するためにソフトウェアアップグレードにより、アップグレードされ得る、または、ソフトウェアプログラム1133は、クラウドサービスからのカレンダーアップデートを要求するように構成され得る。

30

## 【 0 0 8 2 】

週末、複数日の祝祭、および認められた休日の期間を使用する構成において、工程750で決定された時間ギャップの追加的な縮小が、工程760で適用される。一つの構成において、ギャップは、10%の事前に定義された量によりさらに縮小され得る。

## 【 0 0 8 3 】

別の構成では、更なる縮小が、3つの期間（週末、複数日の祝祭、および認められた休日の期間）のうちの選択されたサブセットに対して適用され得る。

## 【 0 0 8 4 】

40

更なる構成では、撮影者に対して個別の追加的な期間が、縮小のために認識され得る。例えば、撮影者が短い休日を取りそうな、結婚記念のような期間が、縮小のために認識され得る。別の例は、撮影者が休暇を取った経験がある期間である。上述したようなギャップにより表される時間間隔は、重なる期間が、撮影者の結婚記念日付近の期間または撮影者が休暇をとると予想される期間の一つ以上に関連するのであれば、さらに縮小され得る。そのような場合では、個人用の期間を記憶するために、個人用のカレンダーが使用され得る。撮影者は、カレンダーに対する関連する情報を明示的に提供することが課される。撮影者に関連するいくつかの情報は、ソフトウェアプログラム1133が実行される機器において、ソーシャルネットワークまたは個人用のカレンダーを含み得る他のソースから取得され得る。

50



## 【産業上の利用可能性】

## 【0085】

説明した構成は、コンピュータおよびデータ処理の産業に対して、特に画像処理に対して、適用可能である。

## 【0086】

上述のものは、本発のいくつかの実施形態についてのみ説明した。修正、および／または変更は、発明および図示し制限されない実施形態の範囲と主旨を離れること無しになされることが可能である。

## 【0087】

本明細書の文脈において、「含む (comprising)」の意味は、「主だったもので、必ずしも単一としないものを含む」、「有する (having)」、「含有する (including)」を意味し、「それだけにより構成される」を意味しない。「含む (comprising)」の言葉の変化、例えば、「comprise」と「comprises」は、対応して変化した意味を有する。

10

【図1】

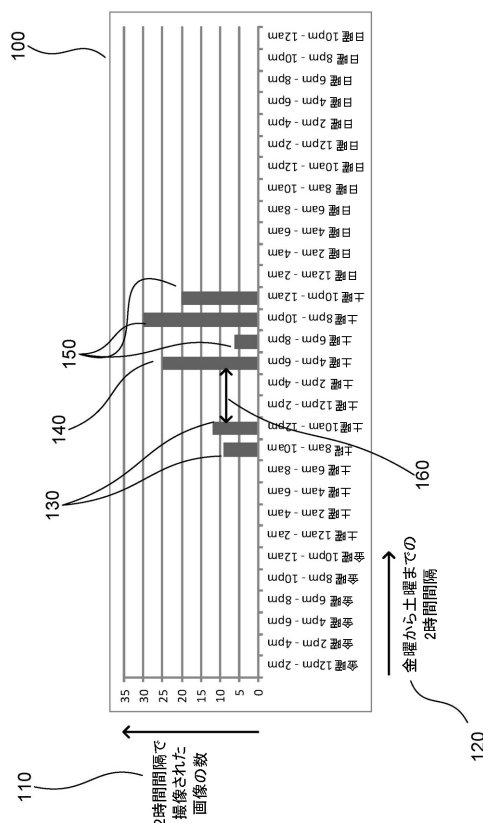


Fig. 1

【図2】

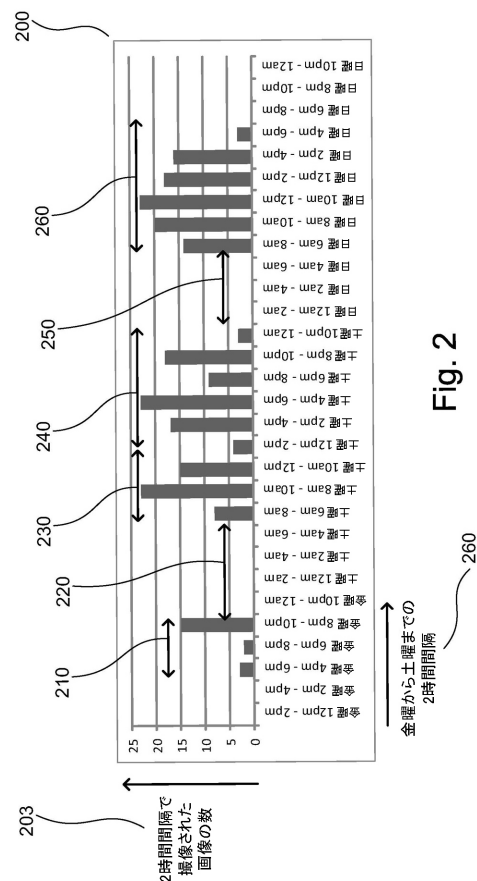


Fig. 2

【図 3】

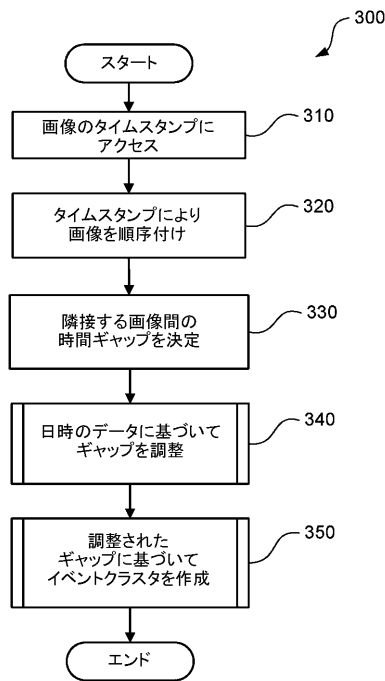


Fig. 3

【図 4】

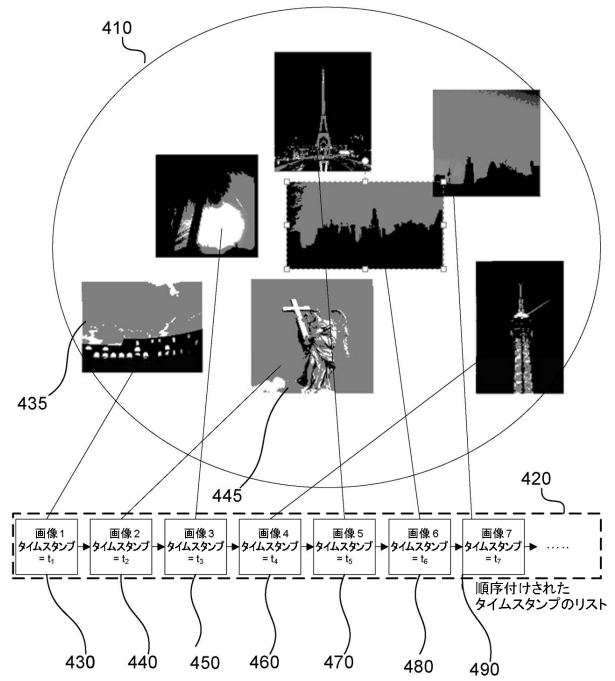


Fig. 4

【図 5】

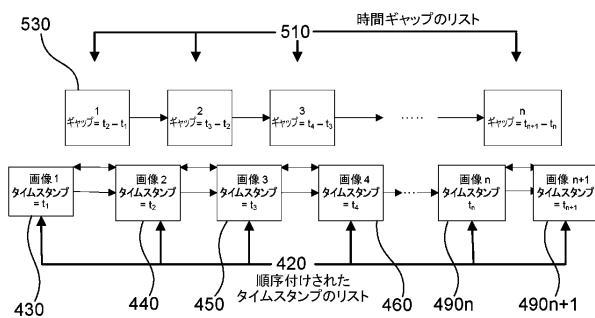


Fig. 5

【図 6】

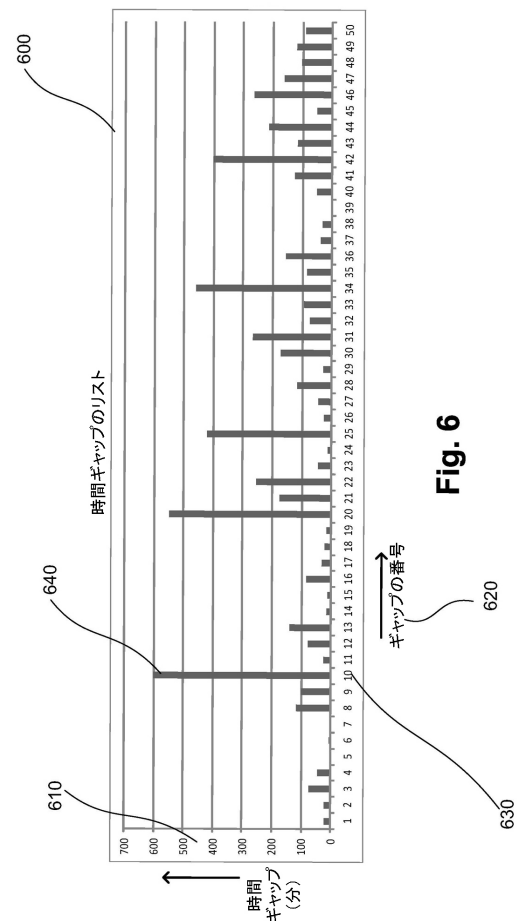


Fig. 6

【図 7】

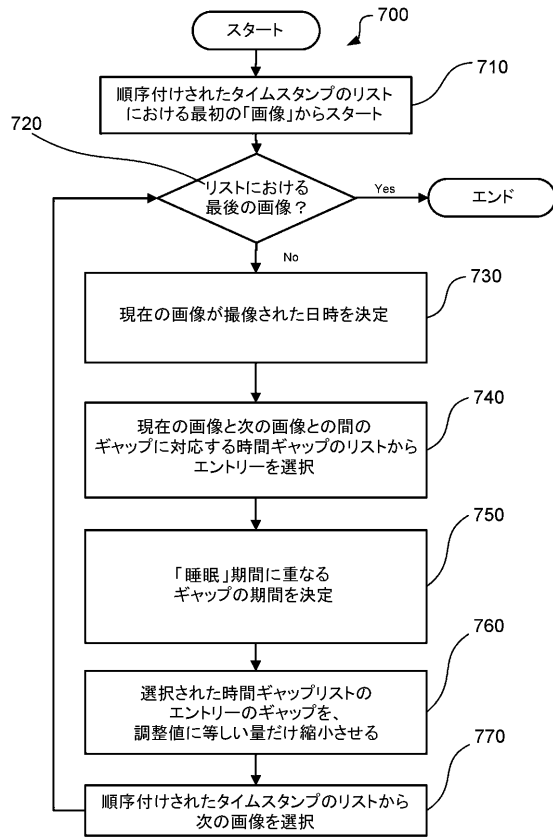


Fig. 7

【図 9】

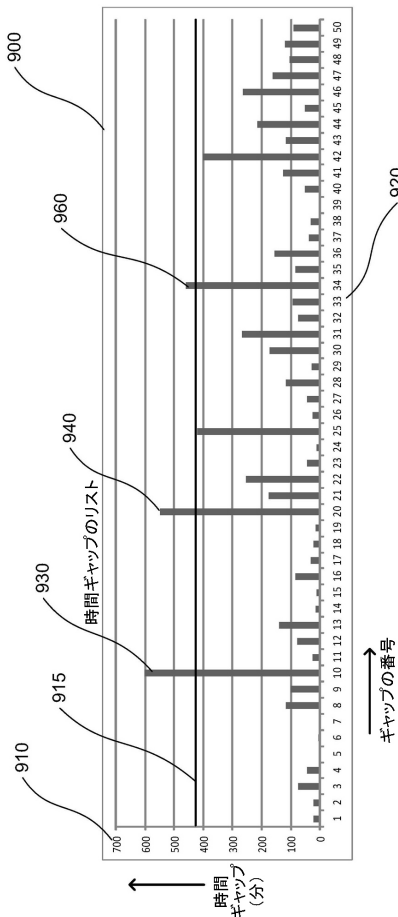


Fig. 9

【図 8】

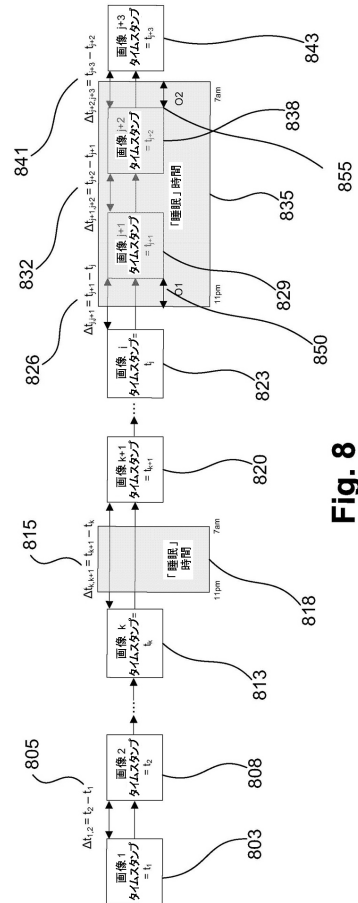


Fig. 8

【図 10】

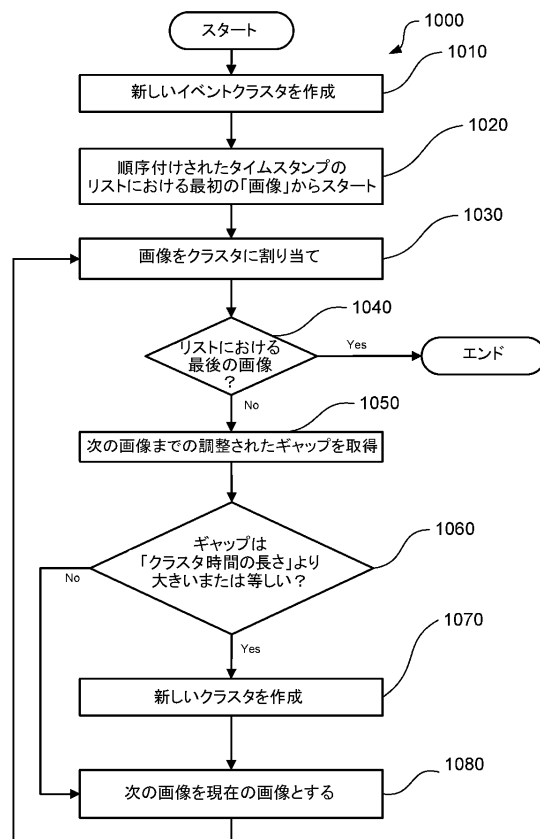


Fig. 10

【図 11A】

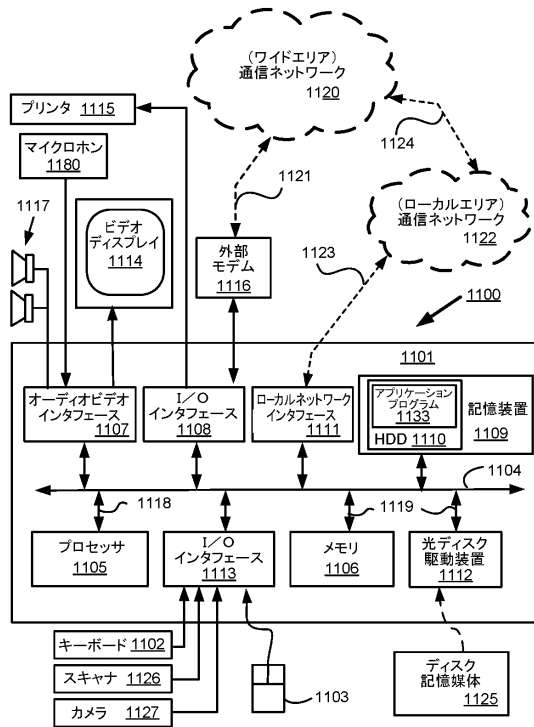


Fig. 11A

【図 11B】

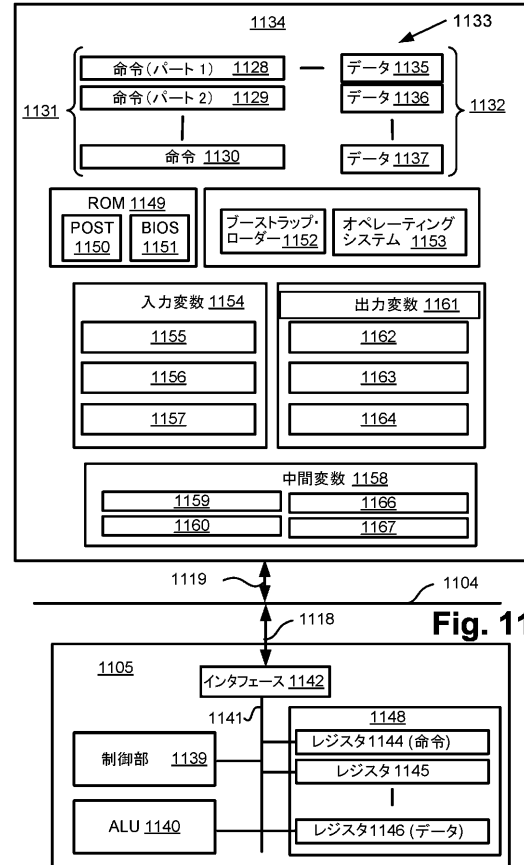


Fig. 11B

## フロントページの続き

(51)Int.Cl.

F I

G 0 6 F 17/30 2 1 0 D

(72)発明者 マーク ロナルド テインシュ

オーストラリア国 2 1 1 3 ニュー サウス ウェールズ州, ノース ライド, トーマス  
ホルト ドライブ 1 キヤノン インフォメーション システムズ リサーチ オーストラリ  
ア プロプライエタリー リミテッド 内

(72)発明者 アレックス ペノフ

オーストラリア国 2 1 1 3 ニュー サウス ウェールズ州, ノース ライド, トーマス  
ホルト ドライブ 1 キヤノン インフォメーション システムズ リサーチ オーストラリ  
ア プロプライエタリー リミテッド 内

(72)発明者 サミー チャン

オーストラリア国 2 1 1 3 ニュー サウス ウェールズ州, ノース ライド, トーマス  
ホルト ドライブ 1 キヤノン インフォメーション システムズ リサーチ オーストラリ  
ア プロプライエタリー リミテッド 内

審査官 川中 龍太

(56)参考文献 特開2007-193592(JP,A)

特開2010-277204(JP,A)

特開2010-118056(JP,A)

特開2007-122110(JP,A)

国際公開第2008/133046(WO,A1)

特開2007-193587(JP,A)

特表2008-521133(JP,A)

特開2006-060518(JP,A)

特開2001-228528(JP,A)

(58)調査した分野(Int.Cl., DB名)

H 0 4 N 5 / 7 6

H 0 4 N 5 / 9 1 - 5 / 9 5 6

H 0 4 N 5 / 2 2 2 - 5 / 2 5 7

G 0 6 F 1 7 / 3 0