

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第5850188号  
(P5850188)

(45) 発行日 平成28年2月3日(2016.2.3)

(24) 登録日 平成27年12月11日(2015.12.11)

(51) Int.Cl. F I  
**HO4N 5/262 (2006.01)** HO4N 5/262

請求項の数 5 (全 38 頁)

(21) 出願番号	特願2015-11594 (P2015-11594)	(73) 特許権者	000002897
(22) 出願日	平成27年1月23日(2015.1.23)		大日本印刷株式会社
(62) 分割の表示	特願2011-13700 (P2011-13700) の分割		東京都新宿区市谷加賀町一丁目1番1号
原出願日	平成23年1月26日(2011.1.26)	(74) 代理人	100122529
(65) 公開番号	特開2015-99603 (P2015-99603A)		弁理士 藤 柝 裕実
(43) 公開日	平成27年5月28日(2015.5.28)	(74) 代理人	100135954
審査請求日	平成27年1月27日(2015.1.27)		弁理士 深町 圭子
(31) 優先権主張番号	特願2010-39211 (P2010-39211)	(74) 代理人	100119057
(32) 優先日	平成22年2月24日(2010.2.24)		弁理士 伊藤 英生
(33) 優先権主張国	日本国(JP)	(74) 代理人	100131369
(31) 優先権主張番号	特願2010-216943 (P2010-216943)		弁理士 後藤 直樹
(32) 優先日	平成22年9月28日(2010.9.28)	(74) 代理人	100164987
(33) 優先権主張国	日本国(JP)		弁理士 伊藤 裕介
		(74) 代理人	100171859
			弁理士 立石 英之

最終頁に続く

(54) 【発明の名称】 画像表示システム

(57) 【特許請求の範囲】

【請求項1】

人物を撮影するカメラと、カメラから送出される撮影映像を合成処理する画像処理装置と、合成処理された合成映像を表示するディスプレイとを備えた画像表示システムであって、

前記画像処理装置は、

映像上の1人以上の人物とコンテンツとの合成のタイミングを定めたシナリオデータを記憶したシナリオデータ記憶手段と、

合成に用いるコンテンツを記憶したコンテンツ記憶手段と、

前記カメラから送出された映像の1つのフレームに撮影されている顔画像を検出し、検出した前記顔画像毎に、顔検出枠の位置・矩形サイズを顔検出枠データとして出力する顔検出手段と、

前記顔検出手段から取得した前記顔検出枠データ毎に、前記顔検出手段から一つ前に処理したフレームから取得した前記顔検出枠データと今回処理したフレームにおいて検出した前記顔検出枠データとの位置・矩形サイズ差を示す評価値を算出し、前記評価値の最も小さい一つ前の前記顔検出枠データを今回の前記顔検出枠データ毎に対応付けるトラッキング手段と、

前記顔検出手段が新規に検出した顔画像毎に顔オブジェクトを生成し、前記トラッキング手段から得られる一つ前と今回の前記顔検出枠データの対応付け結果を参照し、事前に定めた状態遷移表に従い前記顔オブジェクトの状態を閲覧状態に遷移させる状態遷移管理

10

20

手段と、

前記顔検出手段により検出された顔検出枠データを含む顔オブジェクトのうち、前記状態遷移管理手段により閲覧状態または待機状態となっている顔オブジェクトに対して、前記シナリオデータに含まれる人物との対応付けを行うシナリオデータ対応付け手段と、

前記対応付けに従って、前記顔オブジェクトを前記シナリオデータの人物に割り当て、前記顔オブジェクトの顔検出枠データの位置、サイズに合わせて、前記コンテンツの位置、サイズを変更し、前記フレーム上に前記コンテンツを合成した合成画像を生成する合成画像生成手段と、

を備えていることを特徴とする画像表示システム。

【請求項 2】

前記シナリオデータは、コマンドを複数有しており、

前記各フレームについて、フレームから人物の抽出を行い、前記顔検出枠データの部分を除去して特徴部分を特定し、当該特徴部分と事前に設定された領域との面積比が所定のしきい値より大きい状態が、所定数のフレームについて連続した場合に、閲覧者からの指示があったと判断する指示判定手段をさらに有し、

前記合成画像生成手段は、前記指示判定手段により指示があったとされた場合に、前記コマンドを切り替えた後、前記シナリオデータ対応付け手段による対応付けに従って、前記顔オブジェクトを前記シナリオデータ中の前記切り替え後のコマンドの人物に割り当てるものであることを特徴とする請求項 1 に記載の画像表示システム。

【請求項 3】

前記指示判定手段は、前記人物の抽出を、前記フレームから背景を除去して得られた背景除去画像と、前記フレームに HSV 変換を行って得られた HSV 画像の重複部分を抽出することにより行うことを特徴とする請求項 2 に記載の画像表示システム。

【請求項 4】

前記合成画像生成手段は、前記状態遷移管理手段が閲覧状態または待機状態を検出した顔オブジェクトから順に、前記シナリオデータの人物に割り当てることを特徴とする請求項 1 から請求項 3 のいずれか一項に記載の画像表示システム。

【請求項 5】

前記シナリオデータは、シナリオ再生開始からの経過時間に基づく人物とコンテンツとの対応関係、コンテンツと当該コンテンツの前記コンテンツ記憶手段における所在との対応関係が定義されたものであり、前記合成画像生成手段は、前記経過時間に従って、前記人物に対応するコンテンツを、前記コンテンツ記憶手段から抽出し、合成を行うことを特徴とする請求項 1 から請求項 4 のいずれか一項に記載の画像表示システム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、撮影した映像を加工して表示する技術に関し、特に撮影されている閲覧者の状態に応じて加工した映像を表示する技術に関する。

【背景技術】

【0002】

ディスプレイやプロジェクタなどの表示装置を用いて広告を表示する広告媒体であるデジタルサイネージ (Digital Signage) が、様々な場所に設置され始めている。デジタルサイネージを用いることで、動画や音声を用いた豊かなコンテンツの提供が可能になるばかりか、デジタルサイネージの設置場所に応じた効率的な広告配信が可能になるため、今後、デジタルサイネージのマーケット拡大が期待されている。

【0003】

最近では、デジタルサイネージについて、様々な改良が施されており、デジタルサイネージの前に存在する閲覧者の動きに応じて表示させる画像を変化させる技術が提案されて

10

20

30

40

50

いる（特許文献1参照）。

【0004】

上記特許文献1に記載の技術では、人の認識情報と動き情報を基に合成画像を生成するが、トラッキング処理を行っていないために、各人の閲覧時間が把握できない。そのため、閲覧時間を基にしたシナリオを実現することは不可能である。加えて、IDによる個別管理もしていないことにより、全ての閲覧者に対して一辺倒の合成しかできず、各個人にインタラクティブに対応して表示するには不十分であった。また、上記特許文献1に記載の技術のように、人の動きを検出するのは、処理負荷が高く、迅速な表示変化を行うことができないという問題もあった。

【先行技術文献】

【特許文献】

【0005】

【特許文献1】特許第4238371号公報

【発明の概要】

【発明が解決しようとする課題】

【0006】

上記特許文献1に記載の技術では、人の認識情報と動き情報を基に合成画像を生成するが、トラッキング処理を行っていないために、各人の閲覧時間が把握できない。そのため、閲覧時間を基にしたシナリオを実現することは不可能である。加えて、IDによる個別管理もしていないことにより、全員に対して一辺倒の合成しかできず、各個人にインタラクティブに対応して表示するには不十分である。また、上記特許文献1に記載の技術のように、人の動きを検出するのは、処理負荷が高く、迅速な表示変化を行うことができないという問題がある。

【0007】

そこで、本発明は、閲覧者の閲覧時間を基にシナリオに応じて各個人にインタラクティブに対応して表示するとともに、迅速に表示変化を行うことが可能な画像表示システムを提供することを課題とする。

【課題を解決するための手段】

【0008】

上記課題を解決するため、本発明第1の態様では、人物を撮影するカメラと、カメラから送出される撮影映像を合成処理する画像処理装置と、合成処理された合成映像を表示するディスプレイとを備えた画像表示システムであって、前記画像処理装置は、映像上の1人以上の人物とコンテンツとの合成のタイミングを定めたシナリオデータを記憶したシナリオデータ記憶手段と、合成に用いるコンテンツを記憶したコンテンツ記憶手段と、前記カメラから送出された映像の1つのフレームに撮影されている顔画像を検出し、検出した前記顔画像毎に、顔検出枠の位置・矩形サイズを顔検出枠データとして出力する顔検出手段と、前記顔検出手段から取得した前記顔検出枠データを、他のフレームの顔検出枠データと対応付けるトラッキング手段と、前記顔検出手段により検出された顔検出枠データを含む顔オブジェクトに対して、前記シナリオデータに含まれる人物との対応付けを行うシナリオデータ対応付け手段と、前記対応付けに従って、前記顔オブジェクトを前記シナリオデータの人物に割り当て、前記顔オブジェクトの顔検出枠データの位置、サイズに合わせて、前記コンテンツの位置、サイズを変更し、前記フレーム上に前記コンテンツを合成した合成画像を生成する合成画像生成手段を備えている画像表示システムを提供する。

【0009】

本発明第1の態様によれば、映像上の1人以上の人物とコンテンツとの合成のタイミングを定めたシナリオデータを用意しておき、撮影映像のフレームから顔検出枠を検出し、顔検出枠のトラッキングを行い、得られた顔オブジェクトをシナリオデータの人物に割り当て、顔オブジェクトの顔検出枠データの位置、サイズに合わせて、コンテンツの位置、サイズを変更し、フレーム上にコンテンツを合成するようにしたので、撮影した映像に映っている人物にリアルタイムで合成を行った映像を表示することができる。

10

20

30

40

50

## 【 0 0 1 0 】

また、本発明第2の態様では、本発明第1の態様による画像表示システムにおいて、前記シナリオデータは、コマンドを複数有しており、前記各フレームについて、フレームから人物の抽出を行い、前記顔検出枠データの部分を除去して特徴部分を特定し、当該特徴部分と事前に設定された領域との面積比が所定のしきい値より大きい状態が、所定数のフレームについて連続した場合に、閲覧者からの指示があったと判断する指示判定手段をさらに有し、前記合成画像生成手段は、前記指示判定手段により指示があったとされた場合に、前記コマンドを切り替えた後、前記シナリオデータ対応付け手段による対応付けに従って、前記顔オブジェクトを前記シナリオデータ中の前記切り替え後のコマンドの人物に割り当てるものであることを特徴とする。

10

## 【 0 0 1 1 】

本発明第2の態様によれば、シナリオデータが、複数のコマンドを有しており、各フレームについて、フレームから人物の抽出を行い、顔検出枠データの部分を除去して特徴部分を特定し、その特徴部分と事前に設定された領域との面積比が所定のしきい値より大きい状態が、所定数のフレームについて連続した場合に、閲覧者からの指示があったと判断し、コマンドを切り替えた後、顔オブジェクトをシナリオデータ中の切り替え後のコマンドの人物に割り当てるようにしたので、撮影した映像に映っている人物の指示に従って、当該人物にリアルタイムで合成を行った映像を表示することができる。

## 【 0 0 1 2 】

また、本発明第3の態様では、本発明第2の態様による画像表示システムにおいて、前記指示判定手段は、前記人物の抽出を、前記フレームから背景を除去して得られた背景除去画像と、前記フレームにHSV変換を行って得られたHSV画像の重複部分を抽出することにより行うことを特徴とする。

20

## 【 0 0 1 3 】

本発明第3の態様によれば、フレームからの人物の抽出を、前記フレームから背景を除去して得られた背景除去画像と、前記フレームにHSV変換を行って得られたHSV画像の重複部分を抽出することにより行うようにしたので、閲覧者からの指示を判定する際、人物の特徴である肌色の部分をより正確に抽出して判定することが可能となる。

## 【 0 0 1 4 】

また、本発明第4の態様では、本発明第1から第3のいずれかの態様による画像表示システムにおいて、前記トラッキング手段は、前記顔検出手段から取得した前記顔検出枠データ毎に、前記顔検出手段から一つ前に処理したフレームから取得した前記顔検出枠データと今回処理したフレームにおいて検出した前記顔検出枠データとの位置・矩形サイズ差を示す評価値を算出し、前記評価値の最も小さい一つ前の前記顔検出枠データを今回の前記顔検出枠データ毎に対応付けるものであり、前記顔検出手段が新規に検出した顔画像毎に顔オブジェクトを生成し、前記トラッキング手段から得られる一つ前と今回の前記顔検出枠データの対応付け結果を参照し、事前に定めた状態遷移表に従い前記顔オブジェクトの状態を閲覧状態に遷移させる状態遷移管理手段をさらに有し、前記シナリオデータ対応付け手段は、前記状態遷移管理手段により閲覧状態または待機状態となっている顔オブジェクトを取得し、前記シナリオデータに含まれる人物との対応付けを行うものであることを特徴とする。

30

40

## 【 0 0 1 5 】

本発明第4の態様によれば、検出された顔検出枠データ毎に、一つ前に処理したフレームから取得した顔検出枠データと今回処理したフレームにおいて検出した顔検出枠データとの位置・矩形サイズ差を示す評価値を算出し、評価値の最も小さい一つ前の顔検出枠データを今回の顔検出枠データ毎に対応付けることによりトラッキングを行い、新規に検出された顔画像毎に顔オブジェクトを生成し、トラッキング手段から得られる一つ前と今回の顔検出枠データの対応付け結果を参照し、事前に定めた状態遷移表に従い顔オブジェクトの状態を閲覧状態に遷移させるようにし、閲覧状態または待機状態となっている顔オブジェクトをシナリオデータに含まれる人物と対応付けるようにしたので、ノイズの可能性

50

の高い顔画像には合成を行わず、適正な顔画像のみに合成を行うことが可能となる。

【0016】

また、本発明第5の態様では、本発明第4の態様による画像表示システムにおいて、前記合成画像生成手段が、前記状態遷移管理手段が閲覧状態を検出した顔オブジェクトから順に、前記シナリオデータの人物に割り当ててることを特徴とする。

【0017】

本発明第5の態様によれば、状態遷移管理手段が閲覧状態を検出した顔オブジェクトから順に、シナリオデータの人物に割り当てるようにしたので、検出された人物が複数であった場合でも、各コンテンツを対応する人物に的確に合成することが可能となる。

【0018】

また、本発明第6の態様では、本発明第1から第3のいずれかの態様による画像表示システムにおいて、前記顔検出手段は、出力した顔検出枠データを顔オブジェクトとして設定するものであり、前記トラッキング手段は、前記顔検出手段から取得した前記顔検出枠データ毎に、一つ後のフレームにおける当該顔検出枠データの追跡を行い、同一の顔オブジェクトとして設定するものであり、前記シナリオデータ対応付け手段は、前記顔検出手段により設定された顔オブジェクトを取得し、前記シナリオデータに含まれる人物との対応付けを行うものであることを特徴とする。

【0019】

本発明第6の態様によれば、出力した顔検出枠データを即座に顔オブジェクトとして設定するため、顔オブジェクトを迅速に設定することができ、シナリオデータとの対応付けも迅速に行うことが可能となる。

【0020】

また、本発明第7の態様では、本発明第1から第6のいずれかの態様による画像表示システムにおいて、前記シナリオデータは、シナリオ再生開始からの経過時間に基づく人物とコンテンツとの対応関係、コンテンツと当該コンテンツの前記コンテンツ記憶手段における所在との対応関係が定義されたものであり、前記合成画像生成手段は、前記経過時間に従って、前記人物に対応するコンテンツを、前記コンテンツ記憶手段から抽出し、合成を行うことを特徴とする。

【0021】

本発明第7の態様によれば、シナリオデータが、シナリオ再生開始からの経過時間に基づく人物とコンテンツとの対応関係、コンテンツとそのコンテンツのコンテンツ記憶手段における所在との対応関係を定義しているので、シナリオに従い、対応するコンテンツがタイミング通り人物に合成され、円滑な合成画像の再生が可能となる。

【発明の効果】

【0022】

本発明によれば、シナリオに従って撮影映像を合成処理した映像を、閲覧者の閲覧時間を基にシナリオに応じて各個人にインタラクティブに対応して表示するとともに、迅速に表示することが可能となるという効果を奏する。

【図面の簡単な説明】

【0023】

【図1】本実施形態における画像表示システム1の構成を説明する図。

【図2】画像表示システム1を構成する画像処理装置2のハードウェアブロック図。

【図3】画像処理装置2に実装されたコンピュータプログラムで実現される機能ブロック図。

【図4】画像処理装置2がフレームを解析する処理を説明するフロー図。

【図5】トラッキング処理を説明するためのフロー図。

【図6】顔検出枠データ対応付け処理を説明するためのフロー図。

【図7】本実施形態における状態遷移表を説明する図。

【図8】人体及び顔検出結果を説明するための図。

【図9】画像処理装置2が合成画像を作成する処理を説明するフロー図。

10

20

30

40

50

- 【図10】ターゲットが1人の場合のXML形式のシナリオデータの一例を示す図。  
 【図11】ターゲットが2人の場合のXML形式のシナリオデータの一例を示す図。  
 【図12】顔検出枠を用いたコンテンツの合成の様子を示す図。  
 【図13】ターゲットが1人の場合の合成画像の表示状態を示す図。  
 【図14】ターゲットが2人の場合の合成画像の表示状態を示す図。  
 【図15】タッチボタン対応の場合のXML形式のシナリオデータの一例を示す図。  
 【図16】タッチボタン対応の場合の合成画像の表示状態を示す図。  
 【図17】指示判定処理における画像処理の様子を示す図。  
 【図18】画像処理装置2'に実装されたコンピュータプログラムで実現される機能ブロック図。  
 【図19】顔検出処理及びトラッキング処理を説明するためのフロー図。

10

【発明を実施するための形態】

【0024】

以下、本発明の好適な実施形態について図面を参照して詳細に説明する。図1は、本実施形態における画像表示システム1の構成を説明する図、図2は、画像表示システム1を構成する画像処理装置2のハードウェアブロック図、図3は、画像処理装置2に実装されたコンピュータプログラムで実現される機能ブロック図である。

【0025】

図1で図示したように、画像表示システム1には、液晶ディスプレイ等の表示デバイスであるディスプレイ3が含まれる。このディスプレイ3には、撮影した画像だけでなく、表示領域を分けて広告を表示するようにしても良い。

20

【0026】

この場合、ディスプレイ3を街頭や店舗などに設置することにより、画像表示システム1はデジタルサイネージとしても機能する。画像表示システム1をデジタルサイネージとして機能させる場合、ディスプレイ3で表示する広告映像を制御するサーバが必要となる。

【0027】

ディスプレイ3には、ディスプレイ3で再生されている映像を見ている人物の顔が撮影されるようにアングルが設定され、ディスプレイ3で再生されている広告映像を閲覧している人物を撮影するビデオカメラ4が設置されている。

30

【0028】

このビデオカメラ4で撮影された映像は、USBポートなどを利用して画像処理装置2に入力され、画像処理装置2は、ビデオカメラ4から送信された映像に含まれるフレームを解析し、ディスプレイ3の前にいる人物や、ディスプレイ3で再生されている映像を閲覧した人物の顔を検出し、閲覧者に関するログ(例えば、ディスプレイ3の閲覧時間)を記憶する。

【0029】

図1で図示した画像表示システム1を構成する装置において、ディスプレイ3及びビデオカメラ4は市販の装置を利用できるが、画像処理装置2は、従来技術にはない特徴を備えているため、ここから、画像処理装置2について詳細に説明する。

40

【0030】

画像処理装置2は汎用のコンピュータを利用して実現することができ、汎用のコンピュータと同様なハードウェアを画像処理装置2は備え、図2では、該ハードウェアとして、CPU2a(CPU: Central Processing Unit)と、BIOSが実装されるROM2b(ROM: Read-Only Memory)と、コンピュータのメインメモリであるRAM2c(RAM: Random Access Memory)と、外部記憶装置として大容量のデータ記憶装置2d(例えば、ハードディスク)と、外部デバイス(ビデオカメラ4)とデータ通信するための入出力インターフェース2eと、ネットワーク通信するためのネットワークインターフェース2fと、表示デバイス(ディスプレイ3)に情報を送出手するための表示出力インターフェース2gと、文字入力デバイス2h(例えば、キーボード)と、ポインティングデバイス2i(例え

50

ば、マウス)を記載している。

【0031】

画像処理装置2のデータ記憶装置2dには、CPU2aを動作させるためのコンピュータプログラムが実装され、このコンピュータプログラムによって、画像処理装置2には図3で図示した手段が備えられる。また、データ記憶装置2dは、画像表示システムに必要な様々なデータを格納することが可能となっており、映像上の1人以上の人物とコンテンツとの合成のタイミングを定めたシナリオデータを記憶したシナリオデータ記憶手段、合成に用いるコンテンツを記憶したコンテンツ記憶手段としての役割も果たしている。

【0032】

ここで、コンテンツ記憶手段に格納されているコンテンツについて説明しておく。コンテンツは、撮影された映像のフレームと合成して合成画像を得る際の素材画像である。図12(a)にコンテンツの一例を示す。コンテンツとしては、特に限定されず、様々な形態のものを用いることができるが、図12(a)の例では、コンテンツとしてカツラを用意した場合を示している。このコンテンツは矩形情報(x, y方向の位置、幅、高さ)を有しており、この矩形情報を用いて、撮影されたフレームとの位置合わせが可能になっている。

【0033】

図3で図示したように、画像処理装置2の入力は、ビデオカメラ4によって撮影された映像で、画像処理装置2の出力は、撮影された映像を加工した加工映像である。

【0034】

画像処理装置2には、ビデオカメラ4によって撮影された映像のフレームを解析する手段として、ビデオカメラ4によって撮影された映像のフレームの背景画像を除去する背景除去手段20と、背景除去手段20によって背景画像が除去されたフレームから人物の顔を検出する顔検出手段21と、背景除去手段20によって背景画像が除去されたフレームから人体を検出する人体検出手段22と、顔検出手段21が検出した顔を前後のフレームで対応付けるトラッキング手段23と、パーティクルフィルタなどの動画解析手法を用い、指定された顔画像をフレームから検出する動画解析手段24と、顔検出手段21が新規に検出した顔画像毎に顔オブジェクトを生成し、トラッキング手段23から得られる一つ前と今回の顔検出枠データの対応付け結果を参照し、事前に定めた状態遷移表に従い顔オブジェクトの状態を遷移させ、顔オブジェクトの状態遷移に応じたログを記憶する状態遷移管理手段25と、顔検出手段21により検出され、状態遷移管理手段25により状態遷移された顔オブジェクトと、用意されたシナリオデータの対応付けを行うシナリオデータ対応付け手段83と、ビデオカメラ4によって撮影された映像の各フレームを、シナリオデータに従って加工した合成画像を作成する合成画像作成手段84と、閲覧者が所定の位置に定義されたタッチレスボタンに対して指示を行っているかどうかを判定する指示判定手段85と、指示判定手段85による判定に従ってシナリオデータ中のコマンドの切替を行うコマンド切替手段86を備え、更に、本実施形態では、ディスプレイ3を閲覧した人物の属性(年齢や性別)をログデータに含ませるために、顔検出手段21が検出した顔画像から人物の人物属性(年齢や性別)を推定する人物属性推定手段26、状態遷移管理手段25が記憶したログをファイル形式で出力するログファイル出力手段27、加工対象のターゲット(人または場所)をシナリオデータ中に定義する合成ターゲット定義手段80、加工に用いるコンテンツ(画像、音声、CG等)をシナリオデータ中に定義する合成コンテンツ定義手段81、加工内容をシナリオデータ中に定義するアニメーションシナリオ定義手段82を備えている。

【0035】

シナリオデータは、別のシステムで事前に作成し、シナリオデータ記憶手段としてのデータ記憶装置2dに格納しておくことができるが、合成ターゲット定義手段80、合成コンテンツ定義手段81、アニメーションシナリオ定義手段82により、作成することもできる。合成ターゲット定義手段80、合成コンテンツ定義手段81、アニメーションシナリオ定義手段82は、撮影により得られた映像の各フレームをどのように加工するかを示

10

20

30

40

50

したシナリオデータを作成するために用いられるものである。シナリオデータの形式は特に限定されないが、本実施形態では、XML (eXtensible Markup Language) を採用している。シナリオデータとしてXMLを採用した本実施形態では、合成ターゲット定義手段80、合成コンテンツ定義手段81、アニメーションシナリオ定義手段82は、テキストエディタで実現することができる。したがって、テキストエディタを起動し、管理者が文字入力デバイスを用いて、文字入力を行うことにより、シナリオデータが作成される。

#### 【0036】

図10は、ターゲットが1人の場合のXML形式のシナリオデータの一例を示す図である。ここからは、図10のシナリオデータを参照しながら、合成ターゲット定義手段80、合成コンテンツ定義手段81、アニメーションシナリオ定義手段82について詳細に説明する。合成ターゲット定義手段80は、ターゲットID、タイプ、絶対座標、移り変わり設定の4つの属性を設定することによりターゲットを定義する。図10の例では、3行目の<Simulation Targets>と、9行目の</Simulation Targets>の2つのタグで囲まれた範囲に対応する。タイプについては、人と場所の2種を設定可能であるが、図10の例では、<Human>タグを用いて人についてのみ設定している。移り変わり設定は、人に対してのみ設定可能となっており、対応付け済みの人が消失した場合、新たな人に対応付けるかどうかを設定するものである。図10の例では、5行目のタグで、ターゲットID、タイプ、移り変わり設定の3属性を設定しており、ターゲットIDは“0”、タイプは“human(人)”、移り変わり設定(IsTransfer)は“false(設定しない)”となっている。

#### 【0037】

合成コンテンツ定義手段81は、コンテンツID、コンテンツのパス、重なり設定の3つの属性を設定することによりコンテンツを定義する。図10の例では、10行目の<Simulation Contents>と、18行目の</Simulation Contents>の2つのタグで囲まれた範囲に対応する。図10の例では、コンテンツID (ContentsID) が“0”から“6”までの7つのコンテンツについて定義されている。図10に示すように、各コンテンツについて1行単位で、コンテンツのパス (ContentsPath)、重なり設定 (OverlapOrder) が設定される。

#### 【0038】

アニメーションシナリオ定義手段82は、コマンドID、コマンドタイプ、開始キー、終了キー、キータイプ、ターゲットID、コンテンツIDの7つの属性を設定することによりアニメーションシナリオを定義する。図10の例では、19行目の<Animation Commands>と、34行目の</Animation Commands>の2つのタグで囲まれた範囲に対応する。図10の例では、コマンドID (CommandID) が“0”から“6”までの7つのコマンドについて定義されている。図10に示すように、各コマンドについて2行単位で、コマンドタイプ、開始キー、終了キー、キータイプ、ターゲットID、コンテンツIDが設定される。コマンドタイプとは、どのようなフレームを基にどのようなタイプの効果を生じさせるかを示すものであり、レイヤ合成、ブレンド合成、音声再生開始が用意されている。このうち、レイヤ合成、ブレンド合成は、画像合成のタイプを示すものであり、レイヤ合成は、コンテンツを上書き合成するものであり、ブレンド合成は、設定された比率に応じてコンテンツとフレームを透過させて合成するものである。図10の例では、コマンドタイプ (CommandType) として、レイヤ合成 (LayerMontage) が設定されている。開始キー、終了キーは各コマンドの開始時点、終了時点を設定するものである。本実施形態では、シナリオデータの時間を、シナリオ開始時を“0.0”、シナリオ終了時を“1.0”として管理している。したがって、最初に開始するコマンドの開始キー (StartKey) は“0.0”、最後に終了するコマンドの終了キー (EndKey) は“1.0”となる。キータイプとは、開始キー、終了キーの基準とする対象を設定するものであり、own、base、globalの3つが用意されている。ownは各ターゲットIDに対応する顔オブジェクトの閲覧時間を基準とし、baseはターゲットID=0に対応する顔オブジェクトの閲覧時間を基準とし、globalは撮影映像の最初のフレームを取得した時間を基準とする。図10の例では、キータイプ (KeyType) として、ownが設定されているので、顔オブジェクトがフレーム

に登場した時点（顔オブジェクトが“閲覧開始”と判断された時点）を“0.0”として、開始キー、終了キーが認識されることになる。

【0039】

シナリオデータの<CycleInterval>タグは、シナリオの開始から終了までの時間を秒単位で設定するものであり、図10の例では、1行目の<CycleInterval>タグにおいて“10”が設定されているので、シナリオの開始から終了まで10秒であることを示している。開始キー、終了キーの値を10倍した実時間でシナリオは管理されることになる。シナリオデータの<IsAutoLoop>タグは、ループ処理（繰り返し処理）を行うかどうかを設定するものであり、図10の例では、2行目の<IsAutoLoop>タグにおいて“true”が設定されているので、ループ処理を行うことを示している。<CycleInterval>タグおよび<IsAutoLoop>タグについても、テキストエディタにより設定が可能である。このようにして、合成ターゲット定義手段80、合成コンテンツ定義手段81、アニメーションシナリオ定義手段82により作成されたシナリオデータは、シナリオデータ記憶手段としてのデータ記憶装置2dに格納される。

10

【0040】

画像処理装置2が、ビデオカメラ4によって撮影された映像のフレームを時系列で解析することで、画像処理装置2のデータ記憶装置2dには、閲覧測定に利用可能なログファイルとして、ディスプレイの閲覧時間が記憶される閲覧時間ログファイルと、ディスプレイを閲覧した人物の位置が記憶される位置ログファイルと、ディスプレイを閲覧した人物の人物属性（例えば、年齢・性別）が記憶される人物属性ログファイルと、ディスプレイの前のいる人物の総人数、ディスプレイを閲覧していない人物の人数、ディスプレイを閲覧した人物の人数が記憶される人数ログファイルが記憶され、これらのログファイルを出力するログファイル出力手段27が画像処理装置2には備えられている。本発明では、ログファイルを作成することは必須ではないが、ログファイルを作成する過程における顔オブジェクト、閲覧開始時刻が、合成画像の生成に利用される。

20

【0041】

まず、ビデオカメラ4から送信された映像のフレームを画像処理装置2が解析する処理を説明しながら、ビデオカメラ4によって撮影された映像のフレームを解析、加工するために備えられた各手段について説明する。

【0042】

図4は、ビデオカメラ4から送信された映像のフレームを画像処理装置2が解析する処理を説明するフロー図である。それぞれの処理の詳細は後述するが、画像処理装置2に映像の一つのフレームが入力されると、画像処理装置2は該フレームについて背景除去処理S1を行い、背景除去処理S1した後のフレームについて、顔検出処理S2及び人体検出処理S3を行う。

30

【0043】

画像処理装置2は、背景除去処理S1した後のフレームについて、顔検出処理S2及び人体検出処理S3を行った後、顔検出処理S3の結果を利用して、今回の処理対象となるフレームであるNフレームから検出された顔と、一つ前のフレームであるN-1フレームから検出された顔を対応付けるトラッキング処理S4を行い、トラッキング処理S4の結果を踏まえて顔オブジェクトの状態を遷移させる状態遷移管理処理S5を実行する。

40

【0044】

まず、背景除去処理S1について説明する。背景除去処理S1を担う手段は、画像処理装置2の背景除去手段20である。画像処理装置2が背景除去処理S1を実行するのは、図1に図示しているように、ディスプレイ3の上部に設けられたビデオカメラ4の位置・アングルは固定であるため、ビデオカメラ4が撮影した映像には変化しない背景画像が含まれることになり、この背景画像を除去することで、精度よく人体・顔を検出できるようにするためである。

【0045】

画像処理装置2の背景除去手段20が実行する背景除去処理としては既存技術を利用

50

き、ビデオカメラ4が撮影する映像は、例えば、朝、昼、夜で光が変化する場合があるので、背景の時間的な変化を考慮した動的背景更新法を用いることが好適である。

【0046】

背景の時間的な変化を考慮した動的背景更新法としては、例えば、「森田 真司，山澤一誠，寺沢 征彦，横矢 直和："全方位画像センサを用いたネットワーク対応型遠隔監視システム"，電子情報通信学会論文誌(D-II)，Vol. J88-D-II，No. 5，pp. 864-875，(2005.5)」に記載されている手法を用いることができる。

【0047】

次に、画像処理装置2の顔検出手段21によって実行される顔検出処理S2について説明する。顔検出処理S2で実施する顔検出方法としては、特許文献1に記載されているような顔検出方法も含め、様々な顔検出方法が開示されているが、本実施形態では、弱い識別器として白黒のHaar-Like特徴を用いたAdaboostアルゴリズムによる顔検出法を採用している。なお、弱い識別器として白黒のHaar-Like特徴を用いたAdaboostアルゴリズムによる顔検出法については、「Paul Viola and Michael J. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features", IEEE CVPR, 2001.」、「Rainer Lienhart and Jochen Maydt, "An Extended Set of Haar-like Features for Rapid Object Detection", IEEE ICIP 2002, Vol. 1, pp. 900-903, Sep. 2002.」で述べられている。

【0048】

弱い識別器として白黒のHaar-Like特徴を用いたAdaboostアルゴリズムによる顔検出法を実行することで、フレームに含まれる顔画像毎に顔検出枠データが得られ、この顔検出枠データには、顔画像を検出したときに利用した顔検出枠の位置(例えば、左上隅の座標)・矩形サイズ(幅及び高さ)が含まれる。

【0049】

次に、画像処理装置2の人体検出手段22によって実行される人体検出処理S3について説明する。人体を検出する手法としては赤外線センサを用い、人物の体温を利用して人体を検出する手法が良く知られているが、本実施形態では、顔検出処理S2で実施する人体検出方法として、弱い識別器としてHOG(Histogram of Oriented Gradients)特徴を用いたAdaboostアルゴリズムによる人体検出法を採用している。なお、弱い識別器としてHOG(Histogram of Oriented Gradients)特徴を用いたAdaboostアルゴリズムによる人体検出法については、「N. Dalal and B. Triggs, "Histograms of Oriented Gradients", International Conference on Computer Vision, pp. 734-741, 2003.」で述べられている。

【0050】

弱い識別器としてHOG特徴を用いたAdaboostアルゴリズムによる人体検出法を実行することで、フレームに含まれる人体毎に人体検出枠データが得られ、この人体検出枠データには、人体画像を検出したときに利用した人体検出枠の位置(例えば、左上隅の座標)・矩形サイズ(幅及び高さ)が得られる。

【0051】

図8は、人体及び顔検出結果を説明するための図である。図8のフレーム7で撮影されている人物は、人物7a~7fの合計6人が含まれ、画像処理装置2の人体検出手段22はそれぞれの人物7a~7fを検出し、それぞれの人物7a~7fに対応する人体検出枠データ70a~70fを出力する。また、画像処理装置2の顔検出手段21は、両眼が撮影されている人物7a~7cの顔を検出し、それぞれの顔に対応する顔検出枠データ71a~71cを出力する。

【0052】

次に、画像処理装置2のトラッキング手段23によって実行されるトラッキング処理S4について説明する。トラッキング処理S4では、画像処理装置2のトラッキング手段23によって、顔検出手段21がN-1フレームから検出した顔検出枠データと、顔検出手段21がNフレームから検出した顔検出枠データを対応付ける処理が実行される。

【0053】

ここから、画像処理装置2のトラッキング手段23によって実行されるトラッキング処理S4について詳細に説明する。図5は、画像処理装置2のトラッキング手段23によって実行されるトラッキング処理S4を説明するためのフロー図である。

【0054】

画像処理装置2のトラッキング手段23は、Nフレームをトラッキング処理S4するために、まず、Nフレームから得られた顔検出枠データ及び人体検出枠データをそれぞれ顔検出手段21及び人体検出手段22から取得する(S10)。

【0055】

なお、次のトラッキング処理S4において、Nフレームから得られた顔検出枠データは、N-1フレームの顔検出枠データとして利用されるため、画像処理装置2のトラッキング手段23は、Nフレームから得られた顔検出枠データをRAM2cまたはデータ記憶装置2dに記憶する。

10

【0056】

画像処理装置2のトラッキング手段23は、Nフレームの顔検出枠データ及び人体検出枠データを取得すると、Nフレームの人体検出枠データ毎に、ディスプレイの閲覧判定を行う(S11)。

【0057】

上述しているように、人体検出枠データには人体検出枠の位置及び矩形サイズが含まれ、顔検出枠データには顔検出枠の位置及び矩形サイズが含まれるため、顔検出枠が含まれる人体検出枠データは、ディスプレイ3を閲覧している人物の人体検出枠データと判定でき、また、顔検出枠が含まれない人体検出枠データは、ディスプレイ3を閲覧していない人物の人体検出枠データと判定できる。

20

【0058】

画像処理装置2のトラッキング手段23は、このようにして、Nフレームの人体検出枠データ毎にディスプレイの閲覧判定を行うと、Nフレームが撮影されたときの人数ログファイルとして、ディスプレイ3の前にいる人物の総人数、すなわち、人体検出手段22によって検出された人体検出枠データの数と、ディスプレイ3を閲覧していない人物の人数、すなわち、顔検出枠が含まれていない人体検出枠データの数と、ディスプレイ3を閲覧している人物の人数、すなわち、顔検出枠が含まれる人体検出枠データの数を記載した人数ログファイルを生成し、Nフレームのフレーム番号などを付与してデータ記憶装置2dに記憶する。

30

【0059】

画像処理装置2のトラッキング手段23は、Nフレームの人体検出枠データ毎に、ディスプレイの閲覧判定を行うと、顔検出手段21がN-1フレームから検出した顔検出枠データと、顔検出手段21がNフレームから検出した顔検出枠データを対応付ける顔検出枠データ対応付け処理S12を実行する。

【0060】

図6は、顔検出枠データ対応付け処理S12を説明するためのフロー図で、本実施形態では、図6で図示したフローにおいて、以下に記述する数式1の評価関数を用いて得られる評価値を利用して、顔検出枠データの対応付けがなされる。

40

【0061】

なお、数式1の評価関数 $f_1()$ は、ニアレストネイバー法を用いた評価関数で、評価関数 $f_1()$ で得られる評価値は、顔検出枠データの位置・矩形サイズの差を示した評価値になる。また、数式1の評価関数 $f_2()$ で得られる評価値は、評価関数 $f_1()$ から求められる評価値に、顔検出枠データで特定される顔検出枠に含まれる顔画像から得られ、顔画像の特徴を示すSURF特徴量の差が重み付けして加算された評価値になる。

【数 1】

$$f_1(\text{顔検出枠の座標値, 顔検出枠の矩形サイズ}) = W1 \times |\text{顔検出枠の座標値}_n - \text{顔検出枠の座標値}_{n-1}| + W2 \times |\text{顔検出枠のサイズ}_n - \text{顔検出枠のサイズ}_{n-1}|$$

ここで、

$W1, W2, W3$ は、重み係数（任意に指定）

なお、評価値の最小値と他の評価値との差分が閾値以下の場合、

$$f_2(\text{顔検出枠の座標値, 顔検出枠の矩形サイズ, SURF特徴量}) = W1 \times |\text{顔検出枠の座標値}_n - \text{顔検出枠の座標値}_{n-1}| + W2 \times |\text{顔検出枠のサイズ}_n - \text{顔検出枠のサイズ}_{n-1}| + W3 \times |\text{SURF特徴量}_n \text{とSURF特徴量}_{n-1} \text{の差}|$$

なお、下付文字の  $n$  は、 $N$  フレームにおける値を意味し、

下付文字の  $N-1$  は、 $N-1$  フレームにおける値を意味する。

【0062】

$N-1$  フレームから検出した顔検出枠データと  $N$  フレームから検出した顔検出枠データを対応付けるために、画像処理装置 2 のトラッキング手段 23 は、まず、 $N$  フレームから得られた顔検出枠データの数だけループ処理 L1 を実行する。

【0063】

このループ処理 L1 において、画像処理装置 2 のトラッキング手段 23 は、まず、 $N-1$  フレームから検出された顔検出枠データの数だけループ処理 L2 を実行し、このループ処理 L2 では、ループ処理 L1 の処理対象となる顔検出枠データの位置・矩形サイズと、ループ処理 L2 の処理対象となる顔検出枠データの位置・矩形サイズが、数式 1 の評価関数  $f_1()$  に代入して評価値を算出し (S120)、ループ処理 L1 の対象となる顔検出枠データとの位置・サイズの差を示す評価値が、 $N-1$  フレームから検出された顔検出枠データ毎に算出される。

【0064】

画像処理装置 2 のトラッキング手段 23 は、ループ処理 L1 の処理対象となる顔検出枠データとの位置・サイズの差を示す評価値を、 $N-1$  フレームから検出された顔検出枠データ毎に算出すると、該評価値の最小値を検索し (S121)、該評価値の最小値と他の評価値との差分を算出した後 (S122)、閾値以下の該差分値があるか判定する (S123)。

【0065】

そして、画像処理装置 2 のトラッキング手段 23 は、ループ処理 L1 の処理対象となる顔検出枠データとの位置・サイズの差を示す評価値の最小値と他の評価値との差分の中に、閾値以下の差分がある場合、画像処理装置 2 のトラッキング手段 23 は、評価値が閾値以内である顔検出枠データ数だけループ処理 L3 を実行する。

【0066】

このループ処理 L3 では、ループ処理 L1 の処理対象となる顔検出枠データで特定される顔検出枠内の顔画像と、ループ処理 L3 の処理対象となる  $N-1$  フレームの顔検出枠データで特定される顔検出枠内の顔画像との SURF 特徴量の差が求められ、SURF 特徴量の差が数式 1 の評価関数  $f_2()$  に代入され、SURF 特徴量の差を加算した評価値が、 $N-1$  フレームから検出された顔検出枠データ毎に算出される (S124)。

【0067】

数式 1 で示した評価関数  $f_2()$  を用い、SURF 特徴量の差を加算した評価値を算出するのは、ニアレストネイバー法のみを利用した評価関数  $f_1()$  を用いて求められた評価値の最小値と他の評価値との差分値に閾値以下がある場合、サイズの似た顔検出枠が近接していると考えられ (例えば、図 8 の人物 7a, b)、ニアレストネイバー法の評価値

10

20

30

40

50

からでは、Nフレームの顔検出枠データに対応付けるN - 1フレームの顔検出枠データが判定できないからである。

【0068】

数式1で示した評価関数 $f_2()$ を用い、SURF特徴量の差を加算した評価値を算出することで、顔の特徴が加味された評価値が算出されるので、該評価値を用いることで、サイズの似た顔検出枠が近接している場合は、顔が似ているN - 1フレームの顔検出枠データがNフレームの顔検出枠データに対応付けられることになる。

【0069】

そして、画像処理装置2のトラッキング手段23は、数式1の評価関数から得られた評価値が最小値であるN - 1フレームの顔検出枠データを、ループ処理L1の対象となるNフレームの顔検出枠データに対応付ける処理を実行する(S125)。なお、数式1で示した評価関数 $f_2()$ を用いた評価値を算出していない場合、この処理で利用される評価値は、数式1で示した評価関数 $f_1()$ から求められた値になり、数式1で示した評価関数 $f_2()$ を用いた評価値を算出している場合、この処理で利用される評価値は、数式1で示した評価関数 $f_2()$ から求められた値になる。

【0070】

ループ処理L1が終了し、画像処理装置2のトラッキング手段23は、Nフレームの顔検出枠データとN - 1フレームの顔検出枠データに対応付けると、N - 1フレームの顔検出枠データが重複して、Nフレームの顔検出枠データに対応付けられていないか確認する(S126)。

【0071】

N - 1フレームの顔検出枠データが重複して、Nフレームの顔検出枠データに対応付けられている場合、画像処理装置2のトラッキング手段23は、重複して対応付けられているN - 1フレームの顔検出枠データの評価値を参照し、評価値が小さい方を該Nフレームの顔検出枠データに対応付ける処理を再帰的に実行することで、最終的に、Nフレームの顔検出枠データに対応付けるN - 1フレームの顔検出枠データを決定する(S127)。

【0072】

ここから、図4で図示したフローの説明に戻る。トラッキング処理S4が終了すると、画像処理装置2の状態遷移管理手段25によって、トラッキング処理S4から得られ、一つ前と今回の顔検出枠データの対応付け結果を参照し、事前に定めた状態遷移表に従い顔オブジェクトの状態を遷移させ、顔オブジェクトの状態遷移に応じたログを記憶する状態遷移管理処理S5が実行され、この状態遷移管理処理S5で所定の状態遷移があると、該状態遷移に対応した所定のログファイルがデータ記憶装置2dに記憶される。

【0073】

画像処理装置2の状態遷移管理手段25には、顔オブジェクトの状態遷移を管理するために、予め、顔オブジェクトの状態と該状態を状態遷移させるルールが定義された状態遷移表が定められており、画像処理装置2のトラッキング手段23は、この状態遷移表を参照し、顔検出枠データ対応付け処理S12の結果に基づき顔オブジェクトの状態を遷移させる。

【0074】

ここから、状態遷移表の一例を例示し、該状態遷移表の説明をしながら、画像処理装置2の状態遷移管理手段25によって実行される状態遷移管理処理S5について説明する。

【0075】

図7は、本実施形態における状態遷移表6を説明する図である。図7で図示した状態遷移表6によって、顔オブジェクトの状態と、N - 1フレームの状態からNフレームの状態への遷移が定義され、状態遷移表6の縦軸はN - 1フレームの状態で、横軸はNフレームの状態で、縦軸と横軸が交差する箇所に状態遷移する条件が記述されている。なお、状態遷移表に「-」は不正な状態遷移を示している。

【0076】

図7で図示した状態遷移表6には、顔オブジェクトの状態として、None、候補Fa

10

20

30

40

50

ce、現在Face、待機Face、ノイズFace及び終了Faceが定義され、状態遷移表で定義された状態遷移を説明しながら、それぞれの状態について説明する。

【0077】

顔オブジェクトの状態の一つであるNoneとは、顔オブジェクトが存在しない状態を意味し、Nフレームの顔検出枠データに対応付けるN-1フレームの顔検出枠データが無い場合(図7の条件1)、画像処理装置2の状態遷移管理手段25は、顔オブジェクトを識別するためのID、該Nフレームの顔検出データ、顔オブジェクトに付与された状態に係わるデータなどを属性値と有する顔オブジェクトを新規に生成し、該顔オブジェクトの状態を候補Faceに設定する。

【0078】

顔オブジェクトの状態の一つである候補Faceとは、新規に検出した顔画像がノイズである可能性がある状態を意味し、顔オブジェクトの状態の一つに候補Faceを設けているのは、複雑な背景の場合、背景除去処理を行っても顔画像の誤検出が発生し易く、新規に検出できた顔画像がノイズの可能性があるのである。

【0079】

候補Faceの状態である顔オブジェクトには、候補Faceの状態に係わるデータとして、候補Faceの状態であることを示す状態IDと、候補Faceへ状態遷移したときの日時及びカウンタが付与される。

【0080】

候補Faceから状態遷移可能な状態は、候補Face、現在Face及びノイズFaceで、事前に定められた設定時間内において、候補Faceの状態である顔オブジェクトに対応する顔検出枠が所定の数だけ連続してトラッキングできた場合(図7の条件2-2)、該顔オブジェクトの状態は候補Faceから現在Faceに遷移する。

【0081】

候補Faceの状態である顔オブジェクトの属性にカウンタを設けているのは、設定時間内において、候補Faceの状態である顔オブジェクトに対応する顔検出枠を連続してトラッキングできた回数をカウントするため、画像処理装置2の状態遷移管理手段25は、Nフレームの顔検出枠データに対応付けられたN-1フレームの顔検出データが含まれている顔オブジェクトの状態が候補Faceの場合、該顔オブジェクトに付与されている顔検出枠データをNフレームの顔検出枠データに更新すると共に、該顔オブジェクトのカウンタをインクリメントする。

【0082】

そして、画像処理装置2の状態遷移管理手段25は、状態遷移管理処理S5を実行する際、候補Faceである顔オブジェクト毎に、候補Faceへ状態遷移したときの日時を参照し、設定時間以内に該カウンタの値が事前に定めた設定値に達している場合は、顔オブジェクトの状態を現在Faceに状態遷移させる。また、画像処理装置2の状態遷移管理手段25は、この時点で設定時間が経過しているが、該カウンタが設定値に達しなかった該顔オブジェクトの状態をノイズFaceに状態遷移させ(図7の条件2-3)、該設定時間が経過していない該顔オブジェクトについては状態を状態遷移させない(図7の条件2-1)。

【0083】

顔オブジェクトの状態の一つであるノイズFaceとは、画像処理装置2の顔検出手段21が検出した顔画像がノイズと判定された状態で、ノイズFaceに状態遷移した顔オブジェクトは消滅したと見なされ、これ以降の状態遷移管理処理S5に利用されない。

【0084】

顔オブジェクトの状態の一つである現在Faceとは、顔オブジェクトに対応する人物がディスプレイ3を閲覧状態と判定できる状態で、現在Faceの状態にある時間が、顔オブジェクトに対応する人物がディスプレイ3を閲覧している時間となる。

【0085】

画像処理装置2の状態遷移管理手段25は、顔オブジェクトの状態を候補Faceから

10

20

30

40

50

現在 F a c e に状態遷移すると、該顔オブジェクトの顔検出枠データを N フレームの顔検出枠データに更新すると共に、現在 F a c e に係わるデータとして、現在 F a c e の状態であることを示す状態 I D と現在 F a c e に状態遷移させたときの日時を顔オブジェクトに付与する。

【 0 0 8 6 】

また、ディスプレイを閲覧している人物の人物属性（例えば、年齢・性別）をログとして記憶するために、顔オブジェクトの状態を現在 F a c e に状態遷移すると、画像処理装置 2 の状態遷移管理手段 2 5 は人物属性推定手段 2 6 を作動させ、現在 F a c e に状態遷移させた顔オブジェクトの顔検出枠データで特定される顔検出枠から得られる人物属性を取得し、該顔オブジェクトのオブジェクト I D、人物属性が記述された属性ログファイルをデータ記憶装置 2 d に記憶する。

10

【 0 0 8 7 】

なお、画像処理装置 2 に備えられた人物属性推定手段 2 6 については詳細な記載はしないが、人物の顔画像から人物の人物属性（年齢・性別）を自動で識別することは、タバコの自動販売機などでも広く利用されており、例えば、特開 2 0 0 7 - 0 8 0 0 5 7 号公報の技術を利用できる。

【 0 0 8 8 】

更に、画像処理装置 2 の状態遷移管理手段 2 5 は、顔オブジェクトの状態を現在 F a c e に状態遷移すると、ディスプレイ 3 を閲覧している人物の位置を時系列で記憶するための位置ログファイルをデータ記憶装置 2 d に新規に生成する。生成時の位置ログファイルには、現在 F a c e に状態遷移した顔オブジェクトのオブジェクト I D と、現在 F a c e に状態遷移した顔オブジェクトに含まれる顔検出枠データが付与される。

20

【 0 0 8 9 】

現在 F a c e の状態から状態遷移可能な状態は、現在 F a c e 及び待機 F a c e である。画像処理装置 2 の状態遷移管理手段 2 5 は、N フレームの顔検出枠データに対応付けられた N - 1 フレームの顔検出データを含む顔オブジェクトの状態が現在 F a c e の場合（条件 3 - 1）、該顔オブジェクトに付与されている顔検出枠データを N フレームにおける顔検出枠データに更新すると共に、該顔検出枠データを、該顔オブジェクトのオブジェクト I D で特定される位置ログファイルに追加する。

【 0 0 9 0 】

また、画像処理装置 2 の状態遷移管理手段 2 5 は、状態遷移管理処理 S 5 を行う際、N フレームの顔検出枠データが対応付けられなかった N - 1 フレームの顔検出枠データが付与されている顔オブジェクトの状態が現在 F a c e の場合、動画解析手段 2 4 を作動させて、動画解析手法により、該 N - 1 フレームの顔検出枠データに対応する顔画像を N フレームから検出する処理を実施する。

30

【 0 0 9 1 】

本実施形態において、画像処理装置 2 の動画解析手段 2 4 は、まず、N フレームの顔検出枠データが対応付けられなかった N - 1 フレームの顔検出枠データと既に対応付けられている N フレームの顔検出枠データの間で、オクルージョン状態の判定を行い、対象となる人物の顔が完全に隠れた状態のオクルージョンであるか確認する。

40

【 0 0 9 2 】

画像処理装置 2 の動画解析手段 2 4 は、この時点で存在し、現在 F a c e、候補 F a c e 及び待機 F a c e の状態である全ての顔オブジェクトについて、数式 2 に従い、顔オブジェクトのオクルージョン状態を判定する処理を実行する。

## 【数 2】

$$F = (X\_a < X\_b + Width\_b) \&\& (Y\_a < Y\_b + Height\_b) \\ \&\& (X\_a + Width\_a > X\_b) \&\& (Y\_a + Height\_a > Y\_b)$$

$$G = |X\_a - X\_b| + |Y\_a - Y\_b|$$

ここで、

$(X\_a, Y\_a)$ は、顔オブジェクトAに含まれる顔検出枠データの座標

$(X\_b, Y\_b)$ は、顔オブジェクトBに含まれる顔検出枠データの座標

$Width\_a$ 及び $Height\_a$ は、顔検出枠データAに含まれる矩形サイズの幅・高さ

$Width\_b$ 及び $Height\_b$ は、顔検出枠データBに含まれる矩形サイズの幅・高さ

- ・判定基準 1：Fの結果が真になった顔オブジェクトA  
→完全なオクルージョンの可能性があると判定する
- ・判定基準 2：Fの結果が偽かつGの値が閾値以下になった顔オブジェクトA  
→オクルージョン間隙、または、背景色に顔、人物と近い色相系が存在する可能性があるとして判定する
- ・判定基準 3：Fの結果が偽かつGの値が閾値より大きくなった顔オブジェクトA  
→その他（オクルージョンがない可能性が高い）として判定する

## 【0093】

画像処理装置2の動画解析手段24は、数式2に従い、顔オブジェクトのオクルージョン状態を判定する処理を実行すると、判定結果に基づき処理を分岐する。

## 【0094】

トラッキング対象である人物が完全に隠れた状態のオクルージョンである可能性が高いと判断できた場合（数式2の判定基準1に該当する場合）、パーティクルフィルタによるトラッキングを行い、対象となる顔オブジェクトの位置・矩形サイズを検出する。なお、パーティクルフィルタについては、「加藤丈和：「パーティクルフィルタとその実装法」、情報処理学会研究報告、CVIM-157, pp.161-168 (2007).」など数多くの文献で述べられている。

## 【0095】

また、トラッキング対象である人物が半分隠れた状態のオクルージョンで可能性が高いと判断できた場合（数式2の判定基準2に該当する場合）、LK法（Lucas-Kanadeアルゴリズム）によるトラッキングを行い、対象となる顔オブジェクトの位置・矩形サイズを検出する。なお、LK法については、「Lucas, B.D. and Kanade, T.: "An Iterative Image Registration Technique with an Application to Stereo Vision", Proc. DARPA Image Understanding Workshop, pp.121-130, 1981.」で述べられている。

## 【0096】

そして、トラッキング対象である人物にオクルージョンはない可能性が高いと判断できた場合（数式3の判定基準3に該当する場合）、画像処理装置2の動画解析手段24は、CamShift手法を用いたトラッキングを行い、対象となる顔オブジェクトの位置・矩形サイズを検出する。なお、CamShift手法については、「G. R. Bradski: "Computer vision face tracking for use in a perceptual user interface," Intel Technology Journal, Q2, 1998.」で述べられている。

## 【0097】

画像処理装置2の状態遷移管理手段25は、これらのいずれかの手法で対象となる顔画像がNフレームから検出できた場合、現在Faceの状態である顔オブジェクトの顔検出データを、これらの手法で検出された位置・矩形サイズに更新し、これらのいずれかの手法でも対象となる顔画像がトラッキングできなかった場合、現在Faceの状態である顔オブジェクトの状態を待機Faceに状態遷移させる（図7の条件3-2）。

## 【0098】

10

20

30

40

50

顔オブジェクトの状態の一つである待機 F a c e とは、画像処理装置 2 に備えられた動  
画解析手段 2 4 を用いても、顔オブジェクトに対応する顔画像を検出できなくなった状態  
である。

【 0 0 9 9 】

また、画像処理装置 2 の状態遷移管理手段 2 5 は、顔オブジェクトの状態を待機 F a c e  
に状態遷移する際、顔オブジェクトの顔検出枠データは更新せず、待機 F a c e に係わ  
るデータとして、待機 F a c e の状態であることを示す状態 I D と、該顔オブジェクトが  
現在 F a c e に状態遷移したときの日時と、該顔オブジェクトが待機 F a c e に状態遷移  
したときの日時を顔オブジェクトに付与する。

【 0 1 0 0 】

待機 F a c e から状態遷移可能な状態は、現在 F a c e または終了 F a c e である。画  
像処理装置 2 の状態遷移管理手段 2 5 は、待機 F a c e に状態遷移してからの時間が所定  
時間経過する前に、N フレームの顔検出枠データを含む顔オブジェクトを検索し、該顔オ  
ブジェクトの状態が待機 F a c e であった場合、該顔オブジェクトの状態を待機 F a c e  
から現在 F a c e に状態遷移させる（図 7 の条件 4 - 1 ）。

【 0 1 0 1 】

なお、顔オブジェクトの状態を待機 F a c e から現在 F a c e に状態遷移させる際、画  
像処理装置 2 の状態遷移管理手段 2 5 は、該顔オブジェクトが現在 F a c e に状態遷移し  
たときの日時は、待機 F a c e の状態のときに顔オブジェクトに付与されていた該日時を  
利用する。

【 0 1 0 2 】

また、画像処理装置 2 のトラッキング手段 2 3 は、顔オブジェクトの状態遷移を管理す  
る処理を実行する際、待機 F a c e に状態遷移してからの時間が所定時間経過した顔オ  
ブジェクトの状態を終了 F a c e に状態遷移させ（図 7 の条件 4 - 3 ）、該設定時間が経過  
していない該顔オブジェクトについては状態を遷移させない（図 7 の条件 4 - 2 ）。

【 0 1 0 3 】

顔オブジェクトの状態の一つである終了 F a c e とは、画像処理装置 2 が検出できな  
くなった人物に対応する状態で、状態が終了 F a c e になった顔オブジェクトは消滅したと  
見なされ、これ以降の状態遷移管理処理 S 5 で利用されない。

【 0 1 0 4 】

なお、画像処理装置 2 の状態遷移管理手段 2 5 は、顔オブジェクトの状態を終了 F a c e  
に状態遷移する前に、該顔オブジェクトのオブジェクト I D、該顔オブジェクトが現在  
F a c e に状態遷移したときの日時である閲覧開始時刻、該顔オブジェクトが待機 F a c e  
に状態遷移したときの日時である閲覧終了時刻を記述した閲覧時間ログファイルを生成  
しデータ記憶装置 2 d に記憶させる。

【 0 1 0 5 】

以上詳しく説明したように、画像処理装置 2 は、顔検出手段 2 1 が検出した顔毎に生成  
する顔オブジェクトの状態として、None、候補 F a c e、現在 F a c e、待機 F a c e、  
ノイズ F a c e 及び終了 F a c e の 5 つを状態遷移表 6 で定義し、顔オブジェクトに  
対応する顔のトラッキング結果に従い、顔オブジェクトの状態を遷移させることで、顔オ  
ブジェクトの状態遷移に従い、ディスプレイ 3 の閲覧時間をログとして記憶することが可  
能になる。

【 0 1 0 6 】

上述した内容に従えば、顔オブジェクトの状態が現在 F a c e である間は、顔オブジェ  
クトに対応する顔を連続して検出できたことになるため、現在 F a c e の状態にあった時  
間は、ディスプレイ 3 の閲覧時間になる。

【 0 1 0 7 】

また、顔オブジェクトの状態として候補 F a c e を定義しておくことで、ノイズによ  
って顔を誤検出した場合でも、ディスプレイ 3 の閲覧時間への影響はなくなる。また、顔オ  
ブジェクトの状態として待機 F a c e を定義しておくことで、顔を見失った後に、同じ顔

10

20

30

40

50

を検出した場合でも、同じ顔として取り扱うことができるようになる。

#### 【0108】

図9は、ビデオカメラ4から送信された映像のフレームを基に、画像処理装置2が合成画像を作成する処理を説明するフロー図である。画像処理装置2を起動し、使用するシナリオデータを指定すると、まず、シナリオデータ対応付け手段83が、指定されたシナリオデータをデータ記憶装置2dから読み込む(S21)。そして、シナリオデータ対応付け手段83は、シナリオデータを解釈し、シナリオデータに従った画像の作成を開始する(S22)。

#### 【0109】

次に、シナリオデータ対応付け手段83は、状態遷移管理手段25により生成された顔オブジェクトデータを取得する(S23)。顔オブジェクトデータは、オブジェクトID、顔検出枠データ(位置、矩形サイズ)、閲覧時間で構成される。

#### 【0110】

続いて、シナリオデータ対応付け手段83は、状態遷移管理手段25から取得した顔オブジェクトデータをシナリオデータに対応付ける処理を行う(S24)。具体的には、顔オブジェクトデータに含まれる顔検出枠データのオブジェクトIDとターゲットIDを対応付ける。状態遷移管理手段25から複数の顔検出枠データを取得した場合は、候補Faceへ状態遷移したときの日時が最も早いものを“0”に設定し、以降、候補Faceへ状態遷移したときの日時が早い順に“1”“2”“3”と数を1ずつ増加させながら設定していく。図10の例では、シナリオデータには、ターゲットID“0”の1つだけ設定されているので、シナリオデータ対応付け手段83は、ターゲットID“0”が対応付けられたオブジェクトIDで特定される顔検出枠データをターゲットとすることになる。

#### 【0111】

次に、合成画像作成手段84が、フレーム単位で表示用の合成画像を作成する処理を行う(S25)。具体的には、まず、開始時点を時刻“0.0”と設定し、この時刻“0.0”で、シナリオデータの<Animation Commands>を参照する。すると、図10に示すように、開始キー“0.0”から終了キー“0.1”までは、キータイプ“own”、コマンドタイプ“LayerMontage(レイヤ合成)”、ターゲットID“0”、コンテンツID“0”であるので、合成画像作成手段84は、ターゲットID“0”の顔検出枠データに、コンテンツID“0”をレイヤ合成することにより、合成画像を作成することになる。

#### 【0112】

コンテンツID“0”のコンテンツ(合成用素材)は、シナリオデータの<Simulation Contents>を参照することにより、コンテンツのパスが特定できるので、そのパスで特定されるデータ記憶装置2d内の記憶位置からコンテンツを取得する。上述のように、コンテンツには矩形情報が設定されているので、合成画像作成手段84は、この矩形情報と顔検出枠データの矩形サイズが一致するように矩形情報とコンテンツのサイズを変更し、変更したコンテンツを、変更後の矩形情報と顔検出枠データの位置が一致する位置にレイヤ合成を行う。具体的には、図12(b)に示すような顔画像に設定された顔検出枠データの矩形に合わせて、コンテンツに設定された矩形の大きさを変更し、コンテンツに設定された矩形の変更割合に合わせてコンテンツをリサイズし、2つの矩形が一致するように合成処理を行う。この結果、例えば、図12(a)のようにコンテンツがカツラである場合、人の顔に合わせてカツラを被せたような状態の合成画像(図12(c))が得られることになる。合成画像作成手段84は、得られた合成画像を、ディスプレイ3に表示させる。この結果、ディスプレイ3には、撮影映像のフレームに加工が施された合成画像が表示されることになる。

#### 【0113】

1つのフレームについて合成画像の作成を終えたら、シナリオデータ対応付け手段83は、シナリオ中であるかどうかを判断する(S26)。具体的には、シナリオデータに従った画像作成開始からの経過時間でシナリオデータ内の<CycleInterval>タグを参照し、経過時間がシナリオ時間未満である場合は、シナリオ中であると判断し、経過時間がシナ

10

20

30

40

50

リオ時間以上である場合は、シナリオ終了であると判断する。シナリオ中であると判断した場合には、シナリオデータ対応付け手段 8 3 は、S 2 3 に戻って、顔オブジェクトデータを取得する。

【 0 1 1 4 】

そして、S 2 4 において、シナリオデータ対応付け手段 8 3 は、状態遷移管理手段 2 5 から取得した次の顔オブジェクトデータをシナリオデータに対応付ける処理を行う。このときも1回目のループと同様、候補 Face へ状態遷移したときの日時が最も早いものを“ 0 ”に設定し、以降、候補 Face へ状態遷移したときの日時が早い順に“ 1 ” “ 2 ” “ 3 ”と数を1ずつ増加させながら設定していく。そして、シナリオデータに従って、シナリオデータ対応付け手段 8 3 は、ターゲットID “ 0 ” が対応付けられたオブジェクトIDで特定される顔検出枠データをターゲットとする。

10

【 0 1 1 5 】

次に、S 2 5 において、合成画像作成手段 8 4 が、フレーム単位で合成画像を作成する処理を行う。具体的には、経過時間を取得し、取得した経過時間で、シナリオデータの<Animation Commands>を参照する。取得した経過時間が、“ 0 . 1 ”より小さい場合は、上述の場合と同様、図 1 0 のシナリオデータに示すように、開始キー“ 0 . 0 ”から終了キー“ 0 . 1 ”の間であるので、キータイプ“ own ”、コマンドタイプ“ LayerMontage(レイヤ合成)”、ターゲットID “ 0 ”、コンテンツID “ 0 ”であるので、合成画像作成手段 8 4 は、ターゲットID “ 0 ”の顔検出枠データに、コンテンツID “ 0 ”をレイヤ合成することにより、合成画像を作成する。取得した経過時間が、“ 0 . 1 ”を超えていた場合、図 1 0 に示すように、開始キー“ 0 . 1 ”から終了キー“ 0 . 3 ”までは、キータイプ“ own ”、コマンドタイプ“ LayerMontage(レイヤ合成)”、ターゲットID “ 0 ”、コンテンツID “ 1 ”であるので、合成画像作成手段 8 4 は、ターゲットID “ 0 ”の顔検出枠データに、コンテンツID “ 1 ”をレイヤ合成することにより、合成画像を作成することになる。このようにして、S 2 6 においてシナリオ終了であると判断されるまでは、経過時間に従い、シナリオデータを実行する処理を繰り返し行う。

20

【 0 1 1 6 】

S 2 6 において、シナリオ終了であると判断した場合には、シナリオデータ対応付け手段 8 3 は、繰り返して処理を行うかどうかを判断する(S 2 7)。具体的には、シナリオデータ内の<IsAutoLoop>タグを参照し、“ true ”が設定されている場合は、繰り返して処理を行うと判断する。繰り返して処理を行うと判断した場合には、シナリオデータ対応付け手段 8 3 は、経過時間を“ 0 ”にリセットし、経過時間の計測を再び開始するとともに、S 2 2 に戻って、シナリオデータに従った画像の作成を開始する。このように、映像の各フレームから得られた合成画像を順次ディスプレイに表示することにより、加工映像として表示されることになる。

30

【 0 1 1 7 】

図 1 3 は、図 1 0 のシナリオデータに従って得られた合成画像の表示状態を示す図である。まず、フレームにフキダシ「こんにちは」が合成された合成画像が表示される(ア)。シナリオ開始時(StartKey="0.0")から経過時間 1 秒(EndKey="0.1")までのフレームについては、全てフキダシ「こんにちは」(ContentsID="0")が合成された合成画像が表示されることになる。この際、ContentsID="0"で<SimulationContents>を参照することにより、コンテンツのコンテンツ記憶手段内における所在“ fukidashi001.jpg ”を特定し、フキダシ「こんにちは」のコンテンツを取得する。

40

【 0 1 1 8 】

次に、経過時間 1 秒(StartKey="0.1")になると、フキダシ「今日の髪素敵でしょ？」(ContentsID="1")が合成された合成画像が表示される(イ)。経過時間 1 秒(StartKey="0.1")から経過時間 3 秒(EndKey="0.3")までのフレームについては、全てフキダシ「今日の髪素敵でしょ？」が合成された合成画像が表示されることになる。この際、ContentsID="1"で<SimulationContents>を参照することにより、コンテンツのコンテンツ記憶手段内における所在“ fukidashi002.jpg ”を特定し、フキダシ「今日の髪素敵でしょ？」

50

」のコンテンツを取得する。

【 0 1 1 9 】

経過時間 3 秒 (StartKey="0.3") になると、フキダシ「もう少しカジュアルにしようかな」(ContentsID="2") が合成された合成画像が表示される(ウ)。この際、ContentsID="2" で<SimulationContents>を参照することにより、コンテンツのコンテンツ記憶手段内における所在 "fukidashi003.jpg" を特定し、フキダシ「もう少しカジュアルにしようかな」のコンテンツを取得する。

【 0 1 2 0 】

経過時間 3 秒 (StartKey="0.3") から経過時間 6 秒 (EndKey="0.6") までのフレームについては、全てフキダシ「もう少しカジュアルにしようかな」が合成された合成画像が表示されることになる。経過時間 6 秒 (StartKey="0.6") になると、黒いカツラ (Contents ID="5") が合成された合成画像が表示される(エ)。この際、ContentsID="5" で<SimulationContents>を参照することにより、コンテンツのコンテンツ記憶手段内における所在 "afro.jpg" を特定し、黒いカツラのコンテンツを取得する。経過時間 6 秒 (StartKey="0.6") から経過時間 7 秒 (EndKey="0.7") までのフレームについては、全て黒いカツラが合成された合成画像が表示されることになる。

【 0 1 2 1 】

経過時間 7 秒 (StartKey="0.7") になると、フキダシ「こっちがいいかな」(Contents ID="3") と、金髪のカツラ (ContentsID="6") が合成された合成画像が表示される(オ)。この際、ContentsID="6" で<SimulationContents>を参照することにより、コンテンツのコンテンツ記憶手段内における所在 "blond.jpg" を特定し、金髪のカツラのコンテンツを取得する。経過時間 7 秒 (StartKey="0.7") から経過時間 9 秒 (EndKey="0.9") までのフレームについては、全てフキダシ「こっちがいいかな」と、金髪のカツラが合成された合成画像が表示されることになる。経過時間 9 秒 (StartKey="0.9") になると、フキダシ「ねっどうでしょう」(ContentsID="4") と、金髪のカツラが合成された合成画像が表示される(カ)。この際、ContentsID="4" で<SimulationContents>を参照することにより、コンテンツのコンテンツ記憶手段内における所在 "fukidashi005.jpg" を特定し、フキダシ「ねっどうでしょう」のコンテンツを取得する。経過時間 9 秒 (StartKey="0.9") から経過時間 10 秒 (EndKey="1.0") までのフレームについては、全て黒いカツラが合成された合成画像が表示されることになる。

【 0 1 2 2 】

次に、ターゲットが 2 人の場合について説明する。図 11 は、ターゲットが 2 人の場合の XML 形式のシナリオデータの一例を示す図である。図 10 の例と大きく異なるのは、<Simulation Targets>タグで挟まれた<Human>タグ内に、ターゲット ID が 2 つ設定されている点である。図 11 の例では、ターゲット ID "0" と "1" が設定されている。<Animation Commands>タグ内では、各コマンド ID について、1 つのターゲット ID が設定される。図 11 の例では、コマンド ID "0" "2" "3" "5" については、ターゲット ID "0" が設定され、コマンド ID "1" "4" "6" については、ターゲット ID "1" が設定されている。

【 0 1 2 3 】

ターゲットが 2 人の場合も、ターゲットが 1 人の場合と同様に、図 9 のフロー図に従って実行される。ターゲットが 2 人の場合は、S 23 において、シナリオデータ対応付け手段 83 が、状態遷移管理手段 25 により生成された顔オブジェクトデータを複数取得するので、S 24 において、取得したそれぞれの顔オブジェクトデータをシナリオデータに対応付ける。図 11 の例では、シナリオデータには、ターゲット ID "0" "1" の 2 つが設定されているので、シナリオデータ対応付け手段 83 は、ターゲット ID "0" "1" が対応付けられたオブジェクト ID で特定される顔検出枠データをターゲットとすることになる。これにより、ターゲットとする顔オブジェクトの顔検出枠がフレームで入れ替わるように移動したとしても、合成画像作成手段 84 は、それぞれの顔検出枠に合わせてコンテンツを合成することができる。

10

20

30

40

50

## 【 0 1 2 4 】

図 1 4 は、図 1 1 のシナリオデータに従って得られた合成画像の表示状態を示す図である。まず、フレームの左側にフキダシ「こんにちは」が合成された合成画像が表示され、経過時間 1 秒になると、フレームの右側にフキダシ「こんにちは」が合成された合成画像が表示される（ア）。経過時間 3 秒になると、金髪のカツラと、黒いカツラが合成された合成画像が表示される（図示省略）。経過時間 4 秒になると、フキダシ「今日のあなたの髪」と、金髪のカツラと、黒いカツラが合成された合成画像が表示される（イ）。経過時間 6 秒になると、金髪のカツラと、黒いカツラが合成された合成画像が表示される（図示省略）。経過時間 7 秒になると、フキダシ「今日のあなたの髪」と、金髪のカツラと、黒いカツラが合成された合成画像が表示される（ウ）。経過時間 9 秒になると、フキダシ「  
10  
ありがとうあなたもね」と、金髪のカツラと、黒いカツラが合成された合成画像が表示される（エ）。

## 【 0 1 2 5 】

次に、仮想のタッチレスボタンを用いて対話的表示を行う場合について説明する。対話的表示を行うためには、対話的表示に対応するシナリオデータが必要となる。図 1 5 は、タッチレスボタンに対応（ターゲット 1 人）の場合の XML 形式のシナリオデータの一例を示す図である。ここからは、図 1 5 のシナリオデータを参照しながら、合成ターゲット定義手段 8 0、合成コンテンツ定義手段 8 1、アニメーションシナリオ定義手段 8 2 について詳細に説明する。合成ターゲット定義手段 8 0 は、ターゲット ID、タイプ、絶対座標、移り変わり設定の 4 つの属性を設定することによりターゲットを定義する。図 1 5 の例では、1 行目の<Simulation Targets>と、7 行目の</Simulation Targets>の 2 つのタグで囲まれた範囲に対応する。タイプについては、人と場所の 2 種を設定可能であるが、図 1 5 の例では、<Human>タグを用いて人についてのみ設定している。移り変わり設定は、人に対してのみ設定可能となっており、対応付け済みの人が消失した場合、新たな人に対応付けるかどうかを設定するものである。図 1 5 の例では、3 行目のタグで、ターゲット ID、タイプ、移り変わり設定の 3 属性を設定しており、ターゲット ID は “ 0 ”、タイプは “ human(人) ”、移り変わり設定 (IsTransfer) は “ false(設定しない) ” となっている。  
20

## 【 0 1 2 6 】

合成コンテンツ定義手段 8 1 は、コンテンツ ID、コンテンツのパス、重なり設定の 3 つの属性を設定することによりコンテンツを定義する。図 1 5 の例では、8 行目の<Simulation Contents>と、11 行目の</Simulation Contents>の 2 つのタグで囲まれた範囲に対応する。図 1 5 の例では、コンテンツ ID (ContentsID) が “ 0 ” と “ 1 ” の 2 つのコンテンツについて定義されている。図 1 5 に示すように、各コンテンツについて 1 行単位で、コンテンツのパス (ContentsPath)、重なり設定 (OverlapOrder) が設定される。  
30

## 【 0 1 2 7 】

アニメーションシナリオ定義手段 8 2 は、コマンド ID、コマンドタイプ、開始キー、終了キー、キータイプ、ターゲット ID、コンテンツ ID の 7 つの属性を設定することによりアニメーションシナリオを定義する。図 1 5 の例では、12 行目の<Animation Commands>と、17 行目の</Animation Commands>の 2 つのタグで囲まれた範囲に対応する。図 1 5 の例では、コマンド ID (CommandID) が “ 0 ” と “ 1 ” の 2 つのコマンドについて定義されている。図 1 5 に示すように、各コマンドについて 2 行単位で、コマンドタイプ、セクタ ID (SelectorID)、開始キー、終了キー、キータイプ、ターゲット ID、コンテンツ ID が設定される。コマンドタイプとは、どのようなフレームを基にどのようなタイプの効果を生じさせるかを示すものであり、レイヤ合成、ブレンド合成、音声再生開始が用意されている。このうち、レイヤ合成、ブレンド合成は、画像合成のタイプを示すものであり、レイヤ合成は、コンテンツを上書き合成するものであり、ブレンド合成は、設定された比率に応じてコンテンツとフレームを透過させて合成するものである。図 1 5 の例では、コマンドタイプ (CommandType) として、レイヤ合成 (LayerMontage) が設定されている。  
40  
50

## 【 0 1 2 8 】

セレクト ID は、実行するコマンドを特定するための条件であり、そのセレクト ID に対応するコマンド ID のコマンドが実行される。セレクトにより特定された 1 以上のコマンドの集合が 1 つのシナリオを構成することになる。したがって、セレクトとシナリオは 1 対 1 で対応している。図 1 5 の例では、1 つのシナリオが 1 つのコマンドで構成されており、セレクト ID = 0 の場合に、コマンド ID = 0 のコマンドが実行され、セレクト ID = 1 の場合に、コマンド ID = 1 のコマンドが実行されることを示している。開始キー、終了キーは、図 1 0 の例と同様、各コマンドの開始時点、終了時点を設定するものである。本実施形態では、シナリオデータの時間を、シナリオ開始時を “ 0 . 0 ”、シナリオ終了時を “ 1 . 0 ” として管理している。図 1 5 の例では、図 1 0 の例と異なり、全てのコマンドの開始キー (StartKey) は “ 0 . 0 ”、最後に終了するコマンドの終了キー (EndKey) は “ 1 . 0 ” となる。キータイプとは、開始キー、終了キーの基準とする対象を設定するものであり、own、base、global の 3 つが用意されている。own は各ターゲット ID に対応する顔オブジェクトの閲覧時間を基準とし、base はターゲット ID = 0 に対応する顔オブジェクトの閲覧時間を基準とし、global は撮影映像の最初のフレームを取得した時間を基準とする。図 1 0 の例では、キータイプ (KeyType) として、own が設定されているので、顔オブジェクトがフレームに登場した時点 (顔オブジェクトが “ 閲覧開始 ” と判断された時点) を “ 0 . 0 ” として、開始キー、終了キーが認識されることになる。

10

## 【 0 1 2 9 】

図 1 5 の例では、図 1 0 に示した <CycleInterval> タグ、<IsAutoLoop> タグが、2 行目の < Human > と、4 行目の </ Human > の 2 つのタグで囲まれた 3 行目のターゲット ID タグ内に含まれている。<CycleInterval> タグ、<IsAutoLoop> タグがターゲット ID タグ内に含まれている場合でも、図 1 0 の例と同様、シナリオデータの <CycleInterval> タグは、シナリオの開始から終了までの時間を秒単位で設定するものであり、シナリオデータの <IsAutoLoop> タグは、ループ処理 (繰り返し処理) を行うかどうかを設定するものである。図 1 5 の例では、ターゲット ID タグ内で CycleInterval = “ 1 0 ” に設定されているので、シナリオの開始から終了まで 1 0 秒であることを示している。開始キー、終了キーの値を 1 0 倍した実時間でシナリオは管理されることになる。図 1 5 の例では、ターゲット ID タグ内で IsAutoLoop = “ true ” に設定されているので、ループ処理を行うことを示している。このようにして、合成ターゲット定義手段 8 0、合成コンテンツ定義手段 8 1、アニメーションシナリオ定義手段 8 2 により作成されたシナリオデータは、シナリオデータ記憶手段としてのデータ記憶装置 2 d に格納される。

20

30

## 【 0 1 3 0 】

さらに、アニメーションシナリオ定義手段 8 2 は、セレクト ID、セレクトタイプ、セレクトナンバー、トリガ ID の 4 つの属性を設定することによりアニメーションセレクトを定義する。図 1 5 の例では、1 8 行目の <AnimationSelectors SelectorsNum = "2" InitialSelectorID="0"> と、2 1 行目の </AnimationSelectors> の 2 つのタグで囲まれた範囲に対応する。図 1 5 の例では、セレクト ID (SelectorID) が “ 0 ” と “ 1 ” の 2 つのセレクトについて定義されている。図 1 5 に示すように、各セレクトについて 1 行単位で、セレクト ID (SelectorID)、セレクトタイプ (SelectType)、セレクトナンバー (SelectNum)、トリガ ID (TriggerID) が設定される。セレクト ID はセレクトを特定する ID である。セレクトタイプ (SelectType) は、セレクト ID の特定方法を定めるものであり、selectnum、incrementnum、decrementnum、returnnum の 4 つのタイプが存在する。selectnum は、直接指定されたセレクト ID を選択する。incrementnum は、選択中のセレクト ID にセレクトナンバー (SelectNum) を加算して得られるセレクト ID を選択する。decrementnum は、選択中のセレクト ID にセレクトナンバー (SelectNum) を減算して得られるセレクト ID を選択する。returnnum は、トリガ ID に対応するセレクト ID を選択する。セレクトナンバー (SelectNum) は、incrementnum、decrementnum に用いられる番号である。トリガ ID は、所定のイベントが発生した場合に、そのイベントに対応して発生する ID であり、図 1 5 の例では、タッチレスボタンが指示された場合に発生する

40

50

## 【 0 1 3 1 】

さらに、アニメーションシナリオ定義手段 8 2 は、タッチレスボタン ID、ボタンイメージパス、マスクファイルパス、ボタン位置、ボタンサイズ、連続時間しきい値、マスク割合しきい値の 7 つの属性を設定することによりアニメーショントリガを定義する。図 1 5 の例では、2 2 行目の<AnimationTriggers>と、3 1 行目の</AnimationTriggers>の 2 つのタグで囲まれた範囲に対応する。図 1 5 の例では、タッチレスボタン ID (TouchlessButton ID) が “ 0 ” と “ 1 ” の 2 つのボタンについて定義されている。図 1 5 に示すように、各ボタンについて 4 行単位で、タッチレスボタン ID (TouchlessButton ID)、ボタンイメージパス (ButtonImagePath)、マスクファイルパス (MaskFilePath)、ボタン位置 (PositionX, PositionY)、ボタンサイズ (ButtonWidth, ButtonHeight)、連続時間しきい値 (Thitime)、マスク割合しきい値 (Thmask) が設定される。タッチレスボタン ID はタッチレスボタンを特定する ID である。このタッチレスボタン ID はトリガ ID と一対一に対応しており、タッチレスボタン ID が決定すると、トリガ ID も同じ値に設定される。ボタンイメージパスは、ディスプレイ 3 に表示させるタッチレスボタンの画像を記録した位置を特定するパスである。マスクファイルパスは、ボタン形状の角を丸めるためにマスクするためのマスク画像を記録した位置を特定するパスである。ボタン位置は、タッチレスボタンを表示させる画面上の x y 座標である。なお、後述するボタンの指示判定において、ボタン領域を設定する場合は、画像の左右を反転させる必要があるため、それに応じて x 座標を変換して用いる。ボタンサイズは、タッチレスボタンの幅と高さである。連続時間しきい値は、タッチレスボタンが押され続けたと判断するための時間のしきい値である。マスク割合しきい値は、タッチレスボタンが押されていると判断するための面積比率のしきい値である。

## 【 0 1 3 2 】

図 9 のフロー図を用いて、図 1 5 に示したシナリオデータを指定した場合について説明する。画像処理装置 2 を起動し、図 1 5 に示したシナリオデータを使用するシナリオデータとして指定すると、まず、シナリオデータ対応付け手段 8 3 が、指定されたシナリオデータをデータ記憶装置 2 d から読み込む ( S 2 1 )。そして、シナリオデータ対応付け手段 8 3 は、シナリオデータを解釈し、シナリオデータに従った画像の作成を開始する ( S 2 2 )。

## 【 0 1 3 3 】

画像の作成を開始と同時に、画像処理装置 2 は、ディスプレイ 3 にタッチレスボタンのボタン画像を表示する。閲覧者はビデオカメラ 4 により撮影され、常に撮影映像が取得されており、閲覧者がボタン画像に触れようとした際にも、その映像は撮影されることになる。

## 【 0 1 3 4 】

画像の作成を開始したら、シナリオデータ対応付け手段 8 3 は、状態遷移管理手段 2 5 により生成された顔オブジェクトデータを取得する ( S 2 3 )。顔オブジェクトデータは、オブジェクト ID、顔検出枠データ (位置、矩形サイズ)、閲覧時間で構成される。

## 【 0 1 3 5 】

続いて、シナリオデータ対応付け手段 8 3 は、状態遷移管理手段 2 5 から取得した顔オブジェクトデータをシナリオデータに対応付ける処理を行う ( S 2 4 )。具体的には、顔オブジェクトデータに含まれる顔検出枠データのオブジェクト ID とターゲット ID を対応付ける。状態遷移管理手段 2 5 から複数の顔検出枠データを取得した場合は、候補 Face へ状態遷移したときの日時が最も早いものを “ 0 ” に設定し、以降、候補 Face へ状態遷移したときの日時が早い順に “ 1 ” “ 2 ” “ 3 ” と数を 1 ずつ増加させながら設定していく。図 1 5 の例では、シナリオデータには、ターゲット ID “ 0 ” の 1 つだけ設定されているので、シナリオデータ対応付け手段 8 3 は、ターゲット ID “ 0 ” が対応付けられたオブジェクト ID で特定される顔検出枠データをターゲットとすることになる。

## 【 0 1 3 6 】

10

20

30

40

50

次に、合成画像作成手段 8 4 が、フレーム単位で表示用の合成画像を作成する処理を行う ( S 2 5 )。具体的には、まず、開始時点を時刻 “ 0 . 0 ” と設定し、この時刻 “ 0 . 0 ” で、シナリオデータの<Animation Commands>を参照する。図 1 5 に示すように、<Animation Commands>においては、コマンド ID が “ 0 ” と “ 1 ” の 2 つのコマンドが存在し、いずれも、開始キー “ 0 . 0 ” から終了キー “ 1 . 0 ” までは、キータイプ “ own ”、コマンドタイプ “ LayerMontage(レイヤ合成) ”、ターゲット ID “ 0 ” となっており、コンテンツ ID のみが “ 0 ” と “ 1 ” で異なっている。上述のように、コマンド ID は、セレクト ID により決定されるが、図 1 5 の 1 8 行目に示すように、初期セレクト ID ( InitialSelectorID ) は “ 0 ” に設定されているため、最初は、合成画像作成手段 8 4 は、ターゲット ID “ 0 ” の顔検出枠データに、コンテンツ ID “ 0 ” のコンテンツをレイ

10

#### 【 0 1 3 7 】

コンテンツ ID “ 0 ” のコンテンツ ( 合成用素材 ) は、シナリオデータの<Simulation Contents>を参照することにより、コンテンツのパスが特定できるので、そのパスで特定されるデータ記憶装置 2 d 内の記憶位置からコンテンツを取得する。上述のように、コンテンツには矩形情報が設定されているので、合成画像作成手段 8 4 は、この矩形情報と顔検出枠データの矩形サイズが一致するように矩形情報とコンテンツのサイズを変更し、変更したコンテンツを、変更後の矩形情報と顔検出枠データの位置が一致する位置にレイヤ合成を行う。具体的には、図 1 2 ( b ) に示すような顔画像に設定された顔検出枠データの矩形に合わせて、コンテンツに設定された矩形の大きさを変更し、コンテンツに設定された矩形の変更割合に合わせてコンテンツをリサイズし、2 つの矩形が一致するように合成処理を行う。この結果、例えば、図 1 2 ( a ) のようにコンテンツがカツラである場合、人の顔に合わせてカツラを被せたような状態の合成画像 ( 図 1 2 ( c ) ) が得られることになる。合成画像作成手段 8 4 は、得られた合成画像を、ディスプレイ 3 に表示させる。この結果、ディスプレイ 3 には、撮影映像のフレームに加工が施された合成画像が表示されることになる。

20

#### 【 0 1 3 8 】

1 つのフレームについて合成画像の作成を終えたら、シナリオデータ対応付け手段 8 3 は、シナリオ中であるかどうかを判断する ( S 2 6 )。具体的には、シナリオデータに従った画像作成開始からの経過時間でシナリオデータ内の<Target ID>タグ内の “ CycleInterval ” を参照し、経過時間がシナリオ時間未満である場合は、シナリオ中であると判断し、経過時間がシナリオ時間以上である場合は、シナリオ終了であると判断する。シナリオ中であると判断した場合には、シナリオデータ対応付け手段 8 3 は、 S 2 3 に戻って、顔オブジェクトデータを取得する。

30

#### 【 0 1 3 9 】

そして、 S 2 4 において、シナリオデータ対応付け手段 8 3 は、状態遷移管理手段 2 5 から取得した次の顔オブジェクトデータをシナリオデータに対応付ける処理を行う。このときも 1 回目のループと同様、候補 Face へ状態遷移したときの日時が最も早いものを “ 0 ” に設定し、以降、候補 Face へ状態遷移したときの日時が早い順に “ 1 ” “ 2 ” “ 3 ” と数を 1 ずつ増加させながら設定していく。そして、シナリオデータに従って、シナリオデータ対応付け手段 8 3 は、ターゲット ID “ 0 ” が対応付けられたオブジェクト ID で特定される顔検出枠データをターゲットとする。

40

#### 【 0 1 4 0 】

次に、 S 2 5 において、合成画像作成手段 8 4 が、フレーム単位で合成画像を作成する処理を行う。具体的には、経過時間を取得し、取得した経過時間で、シナリオデータの<Animation Commands>を参照する。このとき、セレクト ID が “ 0 ” であれば、コンテンツ ID “ 0 ” のコンテンツを、セレクト ID が “ 1 ” であれば、コンテンツ ID “ 1 ” のコンテンツを、ターゲット ID “ 0 ” の顔検出枠データにレイヤ合成することにより合成画像を作成する。このようにして、 S 2 6 においてシナリオ終了であると判断されるまでは、経過時間に従い、シナリオデータを実行する処理を繰り返し行う。

50

【 0 1 4 1 】

S 2 6において、シナリオ終了であると判断した場合には、シナリオデータ対応付け手段 8 3 は、繰り返して処理を行うかどうかを判断する ( S 2 7 )。具体的には、シナリオデータ内の<IsAutoLoop>タグを参照し、“ true ” が設定されている場合は、繰り返して処理を行うと判断する。繰り返して処理を行うと判断した場合には、シナリオデータ対応付け手段 8 3 は、経過時間を “ 0 ” にリセットし、経過時間の計測を再び開始するとともに、 S 2 2 に戻って、シナリオデータに従った画像の作成を開始する。このように、映像の各フレームから得られた合成画像を順次ディスプレイに表示することにより、加工映像として表示されることになる。

【 0 1 4 2 】

一方、 S 2 2 ~ S 2 5 の処理と並行して、指示判定手段 8 5 は、タッチボタンが指示されたかどうかを常に判断する ( S 2 8 )。具体的には、撮影映像のフレームに対して画像処理を行うことにより、判断を行う。図 1 7 は、指示判定手段 8 5 の画像処理により得られる画像を示す図である。図 1 7 においては、白い部分が画素値 “ 1 ” の画素、黒い部分が画素値 “ 0 ” の画素を示している。まず、指示判定手段 8 5 は、図 4 の S 1 において、背景除去手段 2 0 により処理された背景除去画像を取得する。そして、多値である背景除去画像の各画素値を所定のしきい値で二値化する。図 1 7 ( a ) に、二値化された背景除去画像を示す。続いて、指示判定手段 8 5 は、入力フレーム画像に対して H S V 変換を行い、所定のしきい値で二値化し、二値化された H S V 画像を得る。図 1 7 ( b ) に、二値化された H S V 画像を示す。この H S V 変換により肌色等の色味部分を抽出する。そして、二値化された背景除去画像と二値化された H S V 画像の A n d 演算処理を行う。そして、人物を抽出した人物画像を得る。図 1 7 ( c ) に、人物画像を示す。

【 0 1 4 3 】

次に、指示判定手段 8 5 は、図 4 の S 2 において、顔検出手段 2 1 により取得された顔検出枠を取得する。指示判定手段 8 5 は、取得した顔検出枠をマスクした二値の顔マスク画像を生成する。図 1 7 ( d ) に、顔マスク画像を示す。そして、指示判定手段 8 5 は、顔マスク画像と人物画像の A n d 演算処理を行う。この結果、人物から顔部分を除外した特徴画像が得られる。図 1 7 ( e ) に、特徴画像を示す。

【 0 1 4 4 】

次に、指示判定手段 8 5 は、シナリオデータ中の<AnimationTriggers >と、</AnimationTriggers >の 2 つのタグで囲まれた範囲におけるタッチレスボタンのボタン位置 ( PositionX, PositionY )、ボタンサイズ ( ButtonWidth, ButtonHeight ) の情報を取得する。そして、このボタン位置とボタンサイズから、ボタンの領域、およびボタンに含まれる総画素数を求める。さらに、特徴画像のボタンの領域における画素値 “ 1 ” ( 図面では白で表現 ) の画素数とボタンに含まれる総画素数の比率 R m a s k を算出する。図 1 7 ( f ) に、特徴画像とボタン領域の関係を示す。図 1 7 ( f ) において、白線で囲まれた矩形領域がボタン領域である。なお、図 1 5 に示すように、タッチレスボタンが複数設定されている場合、指示判定手段 8 5 は、各ボタンについて、比率 R m a s k を算出する処理を行う。

【 0 1 4 5 】

指示判定手段 8 5 は、各フレームに対して、上述のような処理を実行し、各タッチレスボタンの比率 R m a s k を算出する。そして、シナリオデータ中に設定されているマスクしきい値 T h m a s k と各タッチレスボタンの比率 R m a s k の比較を行う。そして、比較の結果に基づいて、以下の 1 ) ~ 3 ) のいずれの状態であるかの判断を行う。

【 0 1 4 6 】

( タッチレスボタン上の特徴部分の比率および継続時間による状態の判断 )

- 1 ) R m a s k < T h m a s k の場合 通常状態
- 2 ) R m a s k > T h m a s k かつ T i m e < T h t i m e の場合 手が翳された状態
- 3 ) R m a s k > T h m a s k かつ T i m e > T h t i m e の場合 指示された状態

【 0 1 4 7 】

10

20

30

40

50

上記 2)、3)において、Time は、フレーム数にフレームレートの逆数を乗じて算出した時間であり、Th t i m e はシナリオデータ中に設定されている時間のしきい値である。上記 1)の場合、タッチレスボタン上における手の面積が小さいため、タッチレスボタンに手が翳された状態でなく、通常状態であると判断する。上記 2)の場合、タッチレスボタン上における手の面積が大きいため、手が翳された状態ではあるが、その時間が短いため、タッチレスボタンに対する指示とまでは判断できない。上記 3)の場合、タッチレスボタン上における手の面積が大きく、その時間も長いため、タッチレスボタンに対する指示と判断する。なお、図 15 に示したシナリオデータ中では、2つのタッチレスボタンのいずれにおいても、Th t i m e は 1.0 秒、Th m a s k は 0.1 (10%) が設定されている。

10

## 【0148】

すなわち、S 28 において、指示判定手段 85 は、各タッチレスボタンに対する手の面積比率がしきい値以上であるかどうかを判断する処理を、各フレームについて行い、手の面積比率がしきい値以上であるタッチレスボタンが所定フレーム数以上連続した場合に、タッチレスボタンが指示されたと判断する。

## 【0149】

S 28 において、指示判定手段 85 が、タッチレスボタンが指示されたと判断した場合、コマンド切替手段 86 は、コマンド切替を行う (S 29)。具体的には、セレクト ID (Selector ID) を変更する。図 15 の例では、“0”である場合は“1”に、“1”である場合は“0”に変更されることになる。セレクト ID の変更により、コマンド ID も変更され、コマンドが切り替えられることになる。

20

## 【0150】

図 16 は、図 15 のシナリオデータに従って得られた合成画像の表示状態を示す図である。図 16 (a) (b) において、角の丸い矩形は、タッチレスボタンを示している。図 16 の例では、“金髪”、“黒髪”と記されたタッチレスボタンが表示されている。閲覧者を認識すると、まず、図 15 のシナリオデータ 18 行目に示すように、初期のセレクト ID を定める InitialSelectorID が “0” であるので、セレクト ID は “0” に設定され、13 行目によりコマンド ID が “0” に設定される。そして、13 行目、14 行目の <Command ID> タグに従って、シナリオ開始時 (StartKey="0.0") から経過時間 10 秒 (EndKey="1.0") までのフレームについては、コンテンツ ID “0” のコンテンツがレイヤ合成される。この際、ContentsID="0" で <SimulationContents> を参照することにより、コンテンツのコンテンツ記憶手段内における所在 “afro.jpg” を特定し、黒いカツラのコンテンツを取得する。そして、図 16 (a) に示すように、黒いカツラが閲覧者の画像に合成される。タッチレスボタンの指示によるコマンドの切替がない場合、シナリオ開始時 (StartKey="0.0") から経過時間 10 秒 (EndKey="1.0") までのフレームについては、全て黒いカツラが合成された合成画像が表示されることになる。

30

## 【0151】

閲覧者が “金髪” のタッチレスボタンに手を翳し、<AnimationTriggers> の条件に従って、指示判定手段 85 が、タッチレスボタンが指示されたと判定すると、指示判定手段 85 は、タッチレスボタン ID (TouchlessButton ID) を “1” に設定し、コマンド切替手段 86 が、タッチレスボタン ID に 1 対 1 で対応するトリガ ID を “1” に設定し、さらに、セレクト ID を “1”、コマンド ID を “1” に切り替える。そして、13 行目、14 行目の <Command ID> タグに従って、シナリオ開始時 (StartKey="0.0") から経過時間 10 秒 (EndKey="1.0") までのフレームについては、コンテンツ ID “1” のコンテンツがレイヤ合成される。この際、ContentsID="1" で <SimulationContents> を参照することにより、コンテンツのコンテンツ記憶手段内における所在 “blond.jpg” を特定し、金髪のカツラのコンテンツを取得する。そして、図 16 (b) に示すように、黒いカツラに代えて金髪のカツラが閲覧者の画像に合成される。この後、タッチレスボタンの指示によるコマンドの切替がない場合、シナリオ開始時 (StartKey="0.0") から経過時間 10 秒 (EndKey="1.0") までのフレームについては、全て金髪のカツラが合成された合成画像が表示

40

50

されることになる。

【0152】

< 状態遷移管理手段を用いない構成 >

上記実施形態の画像表示システムは、状態遷移管理手段25を用い、検出された顔画像がノイズであったと判定される場合に、閲覧状態と判断しないようにしたが、状態遷移管理手段25を用いず、検出された顔画像を全て閲覧状態と判断するようにすることも可能である。次に、状態遷移管理手段25を用いない構成について説明する。

【0153】

図18は、状態遷移管理手段25を用いない場合の画像処理装置2'に実装されたコンピュータプログラムで実現される機能ブロック図である。図18において、図3と同一機能を有するものについては、同一符号を付して詳細な説明を省略する。

10

【0154】

図18に示す画像処理装置2'は、図3に示したトラッキング手段23に代えて、トラッキング手段23'を有している。このトラッキング手段23'は、図3に示した動画解析手段24に相当する機能も備えている。

【0155】

図18に示す画像処理装置2'は、フレームを解析するにあたり、図4に示したS1～S5の処理のうち、S1、S3の処理は、画像処理装置2と同様にして行う。また、顔検出処理とトラッキング処理は、連携させて実行する。上述のように、S5の状態遷移管理処理は行わない。

20

【0156】

図19は、顔検出処理とトラッキング処理を示すフロー図である。まず、背景除去処理S1を行った後、Nフレームを処理するにあたり、N-1フレームの顔検出枠の数が0より大であるかどうかの判断を行う(S31)。N-1フレームの顔検出枠の数が0より大である場合は、トラッキング手段23'がトラッキング処理を実行する(S32)。

【0157】

トラッキング手段23'は、N-1フレームにおける各顔検出枠を追跡してNフレームにおける対応する顔検出枠を特定するものである。トラッキング手段23'としては、上述の動画解析手段24が実行する“パーティクルフィルタ”、“LK法”、“CamShift手法”等の公知のトラッキング手法を採用することができる。

30

【0158】

N-1フレームからNフレームへの顔検出枠のトラッキング処理を終えたら、顔検出手段21がNフレームにおける顔検出処理を行う(S33)。S33における顔検出処理は、図4に示したS2の顔検出処理と同一である。また、S31において、N-1フレームの顔検出枠の数が0より大でないと判定された場合は、N-1フレームからNフレームへのトラッキング処理を行わずに、顔検出手段21がNフレームにおける顔検出処理を行う。

【0159】

続いて、顔検出処理S33において新規に検出されたNフレームの顔検出枠の数が0より大であるかどうかを判断する(S34)。新規に検出されたNフレームの顔検出枠とは、Nフレームで検出された顔検出枠のうち、N-1フレームからNフレームへトラッキングされた顔検出枠を除外したものである。

40

【0160】

次に、顔検出手段21が、Nフレームにおいて新規に検出された各顔検出枠データに、オブジェクトIDを付与し、顔検出枠データ、オブジェクトID、トラッキング時間で構成される顔オブジェクトを設定する(S35)。顔オブジェクトは、オブジェクトIDにより特定され、トラッキングにより対応付けられた顔検出枠は、同一のオブジェクトIDで特定されることになる。また、トラッキング時間の初期値は0に設定される。

【0161】

続いて、Nフレームにおける顔検出枠の数が0より大であるかどうかの判断を行う(S

50

36)。S36においては、Nフレームにおいて新規に検出されたかどうかを問わず、既にオブジェクトIDが発行された顔検出枠がNフレームに存在するかどうかを判断する。

【0162】

顔検出枠が存在した場合には、各顔検出枠の顔オブジェクトについて、トラッキング時間を算出する(S37)。具体的には、直前のN-1フレームまでに算出されているトラッキング時間に1フレームに相当する時間を加算することによりNフレームまでの各顔オブジェクトのトラッキング時間を算出する。トラッキング時間を算出し終わったら、Nをインクリメントして(S38)、次のNフレームについての処理に移行する。S36における判断の結果、顔検出枠が存在しなかった場合には、Nフレームには、追跡すべき対象が存在しないことになるので、トラッキング時間の算出は行わず、Nをインクリメントして(S38)、次のNフレームについての処理に移行する。

10

【0163】

画像処理装置2'の顔検出手段21、トラッキング手段23'は、背景除去手段20により背景処理が行われた各フレームについて、図19に示した処理を繰り返し実行する。

【0164】

図19に示した処理において付与された顔オブジェクトは、図9に示したS24において、シナリオデータ対応付け手段83によりシナリオデータと対応付けられる。図19に示した処理においては、顔オブジェクトのオブジェクトIDは、顔検出枠が検出された順に、“0”“1”“2”“3”と数を1ずつ増加させながら設定される。

【0165】

20

以上、本発明の好適な実施形態について説明したが、本発明は上記実施形態に限定されず、種々の変形が可能である。例えば、上記実施形態では、指示判定手段が、人物の抽出を行う際、フレームから背景を除去して得られた背景除去画像と、フレームにHSV変換を行って得られたHSV画像の重複部分を抽出することにより行ったが、撮影されている人物を抽出することができれば、他の手法を用いて抽出しても良い。

【0166】

また、上記実施形態では、シナリオデータに従って合成画像を生成し、表示するようにしたが、音源データを用意しておき、シナリオデータに従って音源データを再生し、スピーカから音を出力するようにしても良い。また、滞在者が数人以上になった場合に、コンテンツを切り替えたり、性別・年代に応じてコンテンツを切り替えたりする処理を実行するようにしても良い。

30

【産業上の利用可能性】

【0167】

本発明は、コンピュータを利用してディスプレイに画像を表示する産業、広告を映像として表示するデジタルサイネージの産業に利用可能である。

【符号の説明】

【0168】

- 1 画像表示システム
- 2、2' 画像処理装置
- 20 背景除去手段
- 21 顔検出手段
- 22 人体検出手段
- 23、23' トラッキング手段
- 24 動画解析手段
- 25 状態遷移管理手段
- 26 人物属性推定手段
- 27 ログファイル出力手段
- 3 ディスプレイ
- 4 ビデオカメラ
- 6 状態遷移表

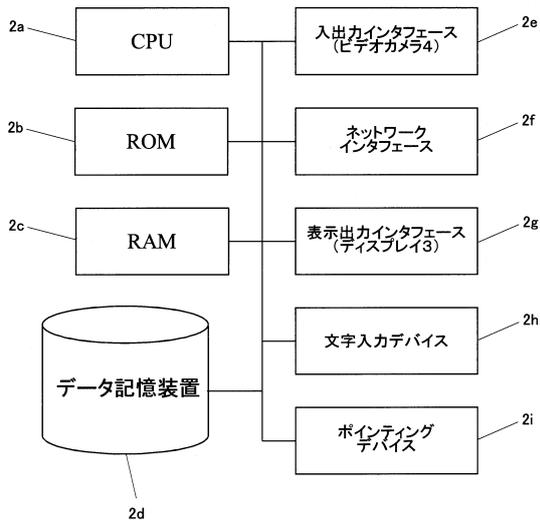
40

50

- 8 0 合成ターゲット定義手段
- 8 1 合成コンテンツ定義手段
- 8 2 アニメーションシナリオ定義手段
- 8 3 シナリオデータ対応付け手段
- 8 4 合成画像作成手段
- 8 5 指示判定手段
- 8 6 コマンド切替手段

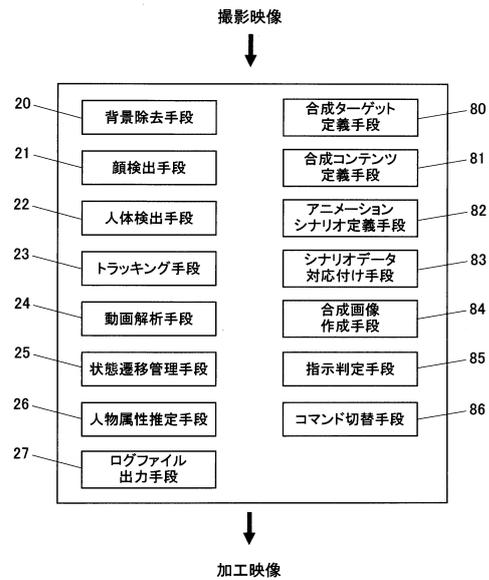
【図2】

ハードウェア構成図(画像処理装置2)

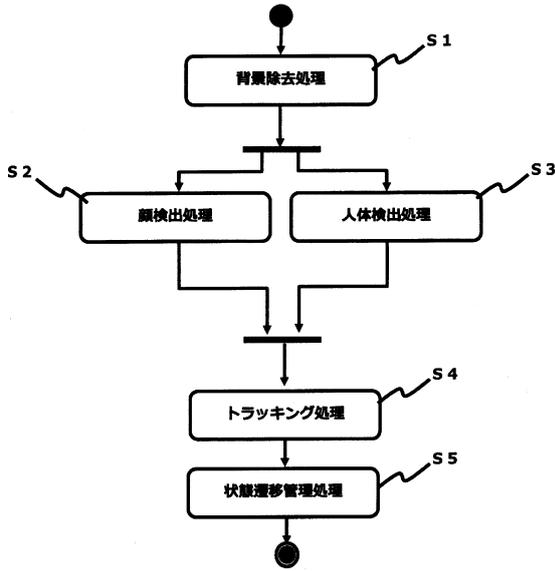


【図3】

機能ブロック図(画像処理装置2)

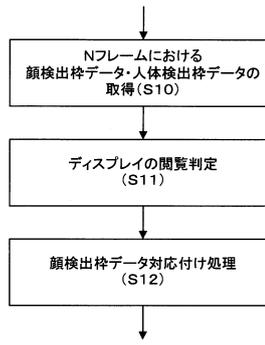


【 図 4 】

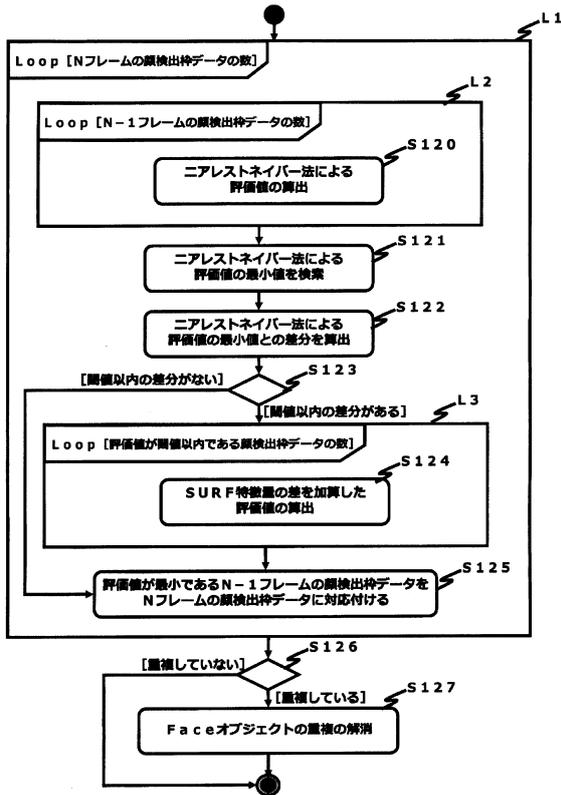


【 図 5 】

トラッキング処理フロー図



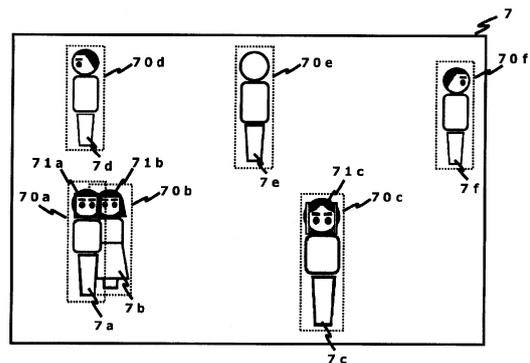
【 図 6 】



【 図 7 】

Nフレーム \ N-1フレーム	候補Face	現在Face	待機Face	ノイズFace	終了Face
None	条件1	-	-	-	-
候補Face	条件2-1	条件2-2	-	条件2-3	-
現在Face	-	条件3-1	条件3-2	-	-
待機Face	-	条件4-1	条件4-2	-	条件4-3
ノイズFace	-	-	-	-	-
終了Face	-	-	-	-	-

【 図 8 】



【 図 1 0 】

シナリオデータ(ターゲット1人)

```

<CycleInterval>10</CycleInterval>
<IsAutoLoop>true</IsAutoLoop>
<SimulationTargets>
  <Human>
    <Target ID="0" Type="human" IsTransfer="false"/>
  </Human>
  <SceneObject>
  </SceneObject>
</SimulationTargets>
<SimulationContents>
  <Contents ID="0" ContentsPath="fukidashi001.jpg" OverlapOrder="4"/>
  <Contents ID="1" ContentsPath="fukidashi002.jpg" OverlapOrder="3"/>
  <Contents ID="2" ContentsPath="fukidashi003.jpg" OverlapOrder="2"/>
  <Contents ID="3" ContentsPath="fukidashi004.jpg" OverlapOrder="1"/>
  <Contents ID="4" ContentsPath="fukidashi005.jpg" OverlapOrder="0"/>
  <Contents ID="5" ContentsPath="afro.jpg" OverlapOrder="6"/>
  <Contents ID="6" ContentsPath="blond.jpg" OverlapOrder="5"/>
</SimulationContents>
<AnimationCommands>
  <Command ID="0" CommandType="LayerMontage"
  StartKey="0.0" EndKey="0.1" KeyType="own" TargetID="0" ContentsID="0"/>
  <Command ID="1" CommandType="LayerMontage"
  StartKey="0.1" EndKey="0.3" KeyType="own" TargetID="0" ContentsID="1"/>
  <Command ID="2" CommandType="LayerMontage"
  StartKey="0.3" EndKey="0.6" KeyType="own" TargetID="0" ContentsID="2"/>
  <Command ID="3" CommandType="LayerMontage"
  StartKey="0.7" EndKey="0.9" KeyType="own" TargetID="0" ContentsID="3"/>
  <Command ID="4" CommandType="LayerMontage"
  StartKey="0.9" EndKey="1.0" KeyType="own" TargetID="0" ContentsID="4"/>
  <Command ID="5" CommandType="LayerMontage"
  StartKey="0.6" EndKey="0.7" KeyType="own" TargetID="0" ContentsID="5"/>
  <Command ID="6" CommandType="LayerMontage"
  StartKey="0.7" EndKey="1.0" KeyType="own" TargetID="0" ContentsID="6"/>
</AnimationCommands>

```

【 図 1 1 】

シナリオデータ(ターゲット2人)

```

<CycleInterval>10</CycleInterval>
<IsAutoLoop>true</IsAutoLoop>
<SimulationTargets>
  <Human>
    <Target ID="0" Type="human" IsTransfer="false"/>
    <Target ID="1" Type="human" IsTransfer="false"/>
  </Human>
  <SceneObject>
  </SceneObject>
</SimulationTargets>
<SimulationContents>
  <Contents ID="0" ContentsPath="fukidashi021.jpg" OverlapOrder="4"/>
  <Contents ID="1" ContentsPath="fukidashi022.jpg" OverlapOrder="3"/>
  <Contents ID="2" ContentsPath="fukidashi023.jpg" OverlapOrder="2"/>
  <Contents ID="3" ContentsPath="fukidashi024.jpg" OverlapOrder="1"/>
  <Contents ID="4" ContentsPath="fukidashi025.jpg" OverlapOrder="0"/>
  <Contents ID="5" ContentsPath="afro.jpg" OverlapOrder="6"/>
  <Contents ID="6" ContentsPath="blond.jpg" OverlapOrder="5"/>
</SimulationContents>
<AnimationCommands>
  <Command ID="0" CommandType="LayerMontage"
  StartKey="0.0" EndKey="0.1" KeyType="own" TargetID="0" ContentsID="0"/>
  <Command ID="1" CommandType="LayerMontage"
  StartKey="0.1" EndKey="0.3" KeyType="base" TargetID="1" ContentsID="1"/>
  <Command ID="2" CommandType="LayerMontage"
  StartKey="0.4" EndKey="0.6" KeyType="own" TargetID="0" ContentsID="2"/>
  <Command ID="3" CommandType="LayerMontage"
  StartKey="0.7" EndKey="0.9" KeyType="own" TargetID="0" ContentsID="3"/>
  <Command ID="4" CommandType="LayerMontage"
  StartKey="0.9" EndKey="1.0" KeyType="base" TargetID="1" ContentsID="4"/>
  <Command ID="5" CommandType="LayerMontage"
  StartKey="0.3" EndKey="1.0" KeyType="own" TargetID="0" ContentsID="5"/>
  <Command ID="6" CommandType="LayerMontage"
  StartKey="0.3" EndKey="1.0" KeyType="base" TargetID="1" ContentsID="6"/>
</AnimationCommands>

```

【 図 1 5 】

シナリオデータ(タッチレスボタン対応:ターゲット1人)

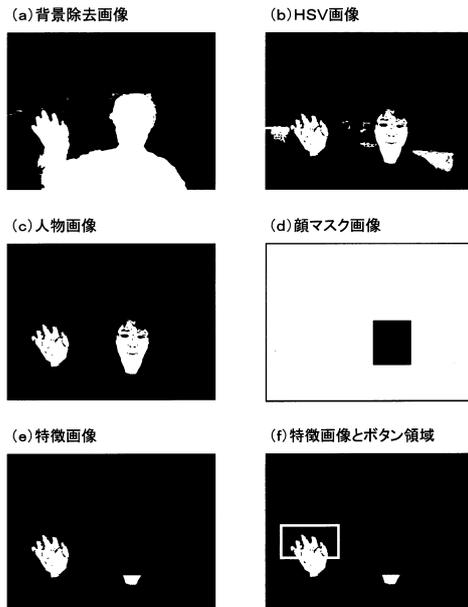
```

<SimulationTargets>
  <Human>
    <Target ID="0" Type="human" IsTransfer="false" CycleInterval="10" IsAutoLoop="true"/>
  </Human>
  <SceneObject>
  </SceneObject>
</SimulationTargets>
<SimulationContents>
  <Contents ID="0" ContentsPath="afro.jpg" OverlapOrder="0"/>
  <Contents ID="1" ContentsPath="blond.jpg" OverlapOrder="1"/>
</SimulationContents>
<AnimationCommands>
  <Command ID="0" CommandType="LayerMontage" SelectorID="0"
  StartKey="0.0" EndKey="1.0" KeyType="own" TargetID="0" ContentsID="0"/>
  <Command ID="1" CommandType="LayerMontage" SelectorID="1"
  StartKey="0.0" EndKey="1.0" KeyType="own" TargetID="0" ContentsID="1"/>
</AnimationCommands>
<AnimationSelectors SelectorsNum="2" InitialSelectorID="0">
  <Selector ID="0" SelectType="returnnum" SelectNum="0" TriggerID="0"/>
  <Selector ID="1" SelectType="returnnum" SelectNum="1" TriggerID="1"/>
</AnimationSelectors>
<AnimationTriggers>
  <TouchlessButton ID="0" ButtonImagePath="/button/button_afro.jpg"
  MaskFilePath="/button/button_afro_mask.jpg"
  PositionX="400" PositionY="100" ButtonWidth="120" ButtonHeight="60" Thtime="1.0"
  Thmask="0.1"/>
  <TouchlessButton ID="1" ButtonImagePath="/button/button_blond.jpg"
  MaskFilePath="/button/button_blond_mask.jpg"
  PositionX="400" PositionY="200" ButtonWidth="120" ButtonHeight="60" Thtime="1.0"
  Thmask="0.1"/>
</AnimationTriggers>

```

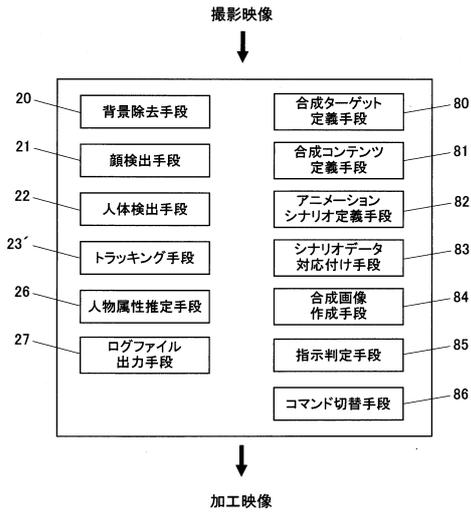
【 図 1 7 】

指示判定手段による特徴部分の抽出



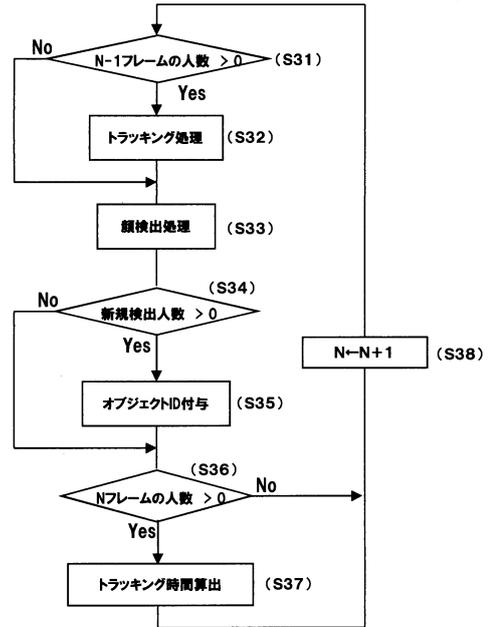
【図18】

機能ブロック図(画像処理装置2')



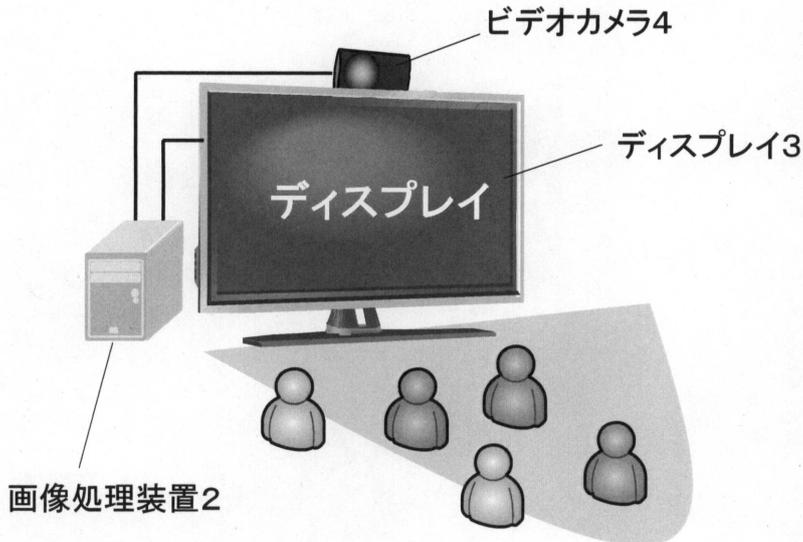
【図19】

顔検出処理+トラッキング処理フロー図



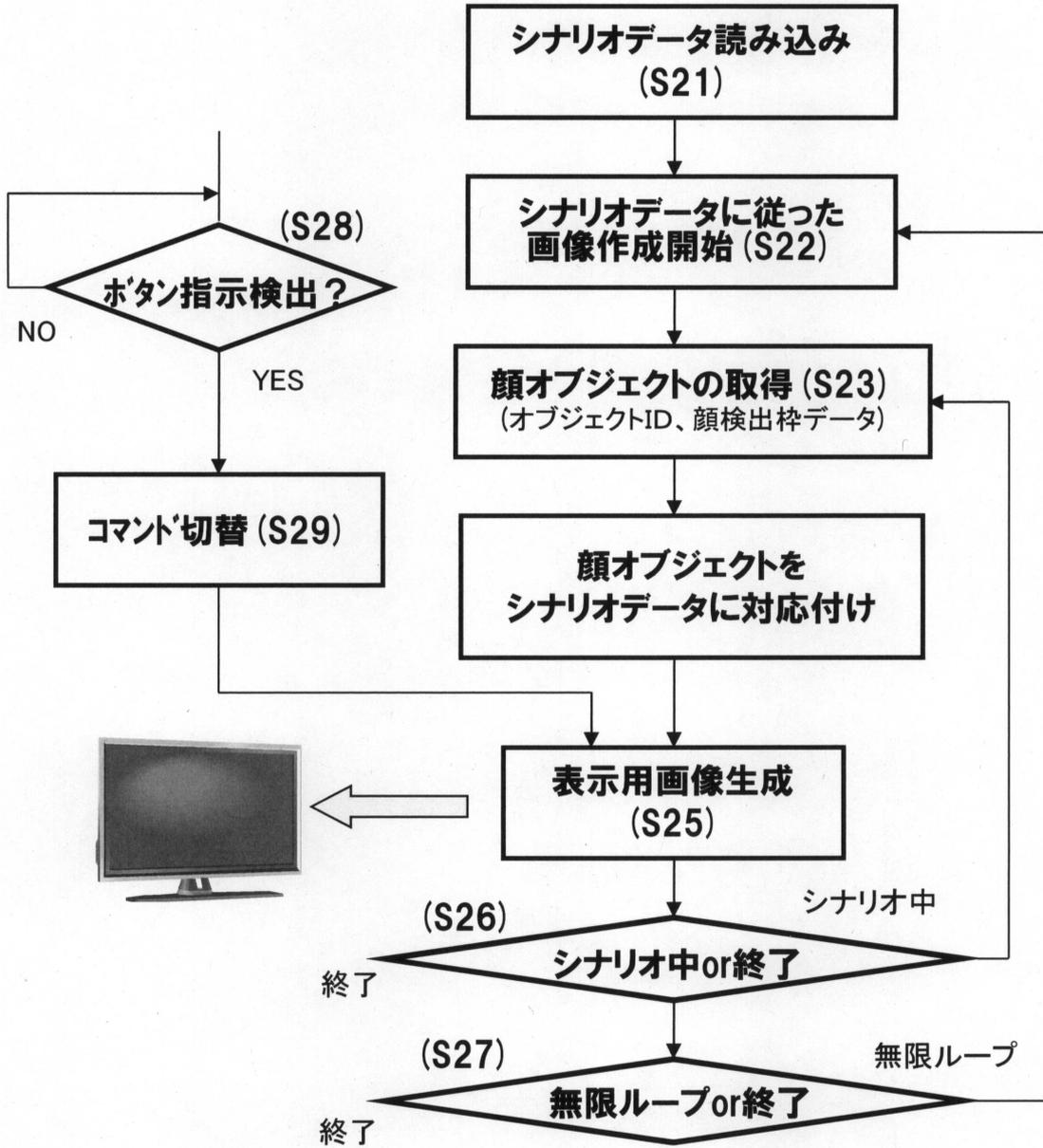
【図1】

画像表示システム1



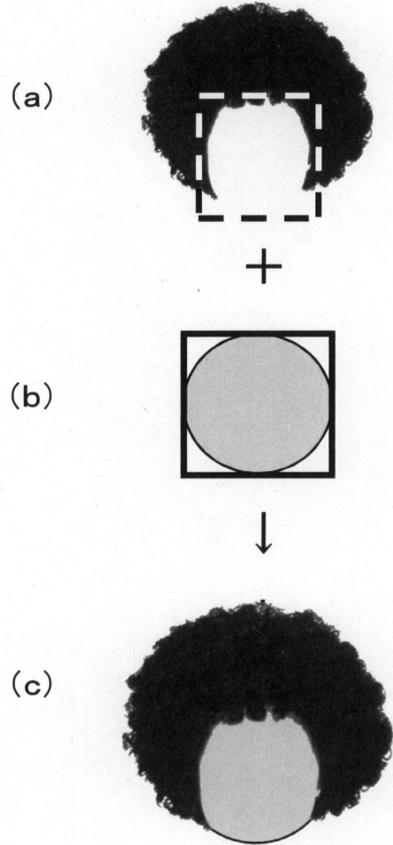
【図9】

画像処理装置フロー図



【図12】

顔検出枠データとコンテンツの合成処理



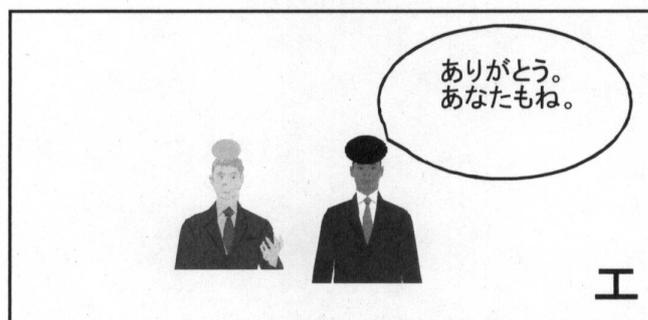
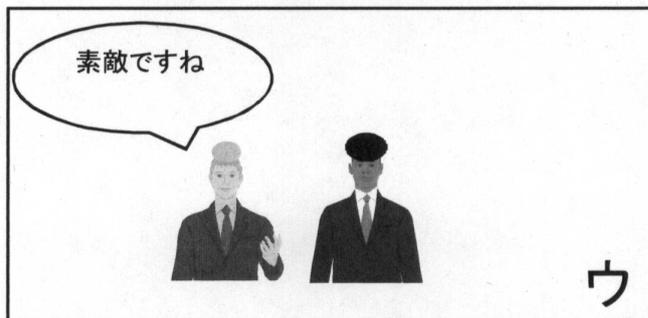
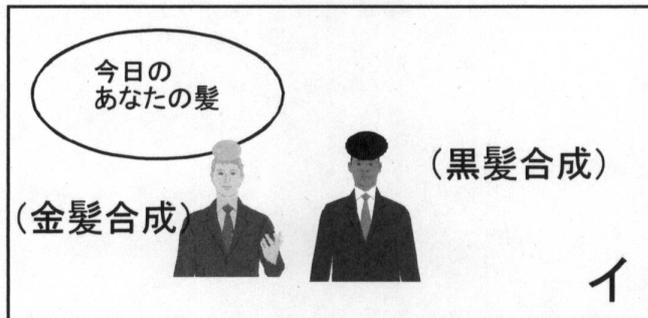
【図13】

合成画像の表示(ターゲット1人)



【図14】

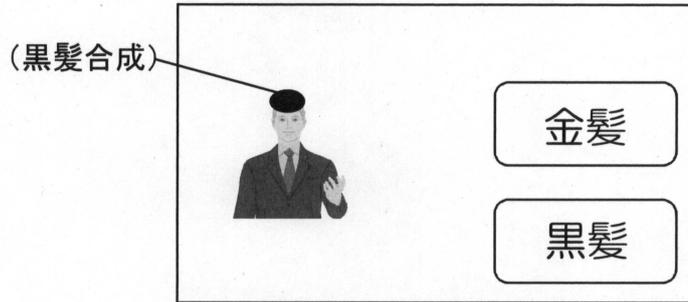
### 合成画像の表示(ターゲット2人)



【図16】

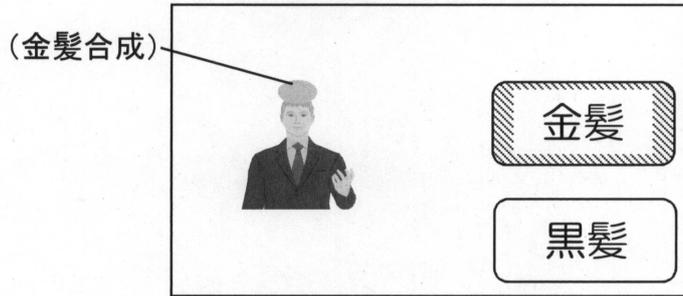
合成画像の表示(タッチレスボタン対応:ターゲット1人)

(a)初回時



初回時、黒髪を合成

(b)コマンド切替時



「金髪」ボタンに手をかざすと、  
金髪を合成

## フロントページの続き

- (72)発明者 田端 聡  
東京都新宿区市谷加賀町一丁目1番1号 大日本印刷株式会社内
- (72)発明者 松葉 靖寿  
東京都新宿区市谷加賀町一丁目1番1号 大日本印刷株式会社内
- (72)発明者 瀬井 大志  
東京都新宿区市谷加賀町一丁目1番1号 大日本印刷株式会社内
- (72)発明者 小泉 和真  
東京都新宿区市谷加賀町一丁目1番1号 大日本印刷株式会社内
- (72)発明者 小野 徹太郎  
東京都新宿区市谷加賀町一丁目1番1号 大日本印刷株式会社内

審査官 佐田 宏史

- (56)参考文献 特開2000-322588(JP,A)  
特開2005-018305(JP,A)  
特開2009-301085(JP,A)  
特開2006-331065(JP,A)  
特開2009-104297(JP,A)  
特開2005-063225(JP,A)  
香山 健太郎、外6名，“プロアクティブ対話型観光案内システムのための顔向き・頭部動作認識”，2009年度人工知能学会全国大会（第23回）論文集，日本，社団法人人工知能学会，2009年 6月17日，pp.1-4

(58)調査した分野(Int.Cl.，DB名)

H04N 5/262, 7/18  
G06T 1/00, 19/00