

修 士 論 文

不特定多数の人物を対象とした 顔変形を含む3次元頭部姿勢の実時間推定

User-Independent Techniques for
Real-Time Estimation of Non-Rigid 3D Head Motion



東京大学大学院
情報理工学系研究科
電子情報学専攻

56425

菅 野 裕 介

指導教員

佐藤 洋一 助教授

平成19年2月

概要

本論文では、ヒューマン・コンピュータ・インタラクション技術への応用を目的として、不特定多数のユーザに対して顔変形および3次元頭部姿勢の実時間推定を実現する手法を提案する。顔変形モデルを用いた姿勢推定をユーザ毎の事前準備を伴うことなく実行するための枠組みとして、本論文では二つの異なる手法を提案する。

第1の手法では、ユーザ頭部の変形モデルの構築と頭部姿勢の実時間推定を同時に実行する。本手法では、複眼カメラによって得られるユーザ頭部の3次元形状を元に、ユーザ独自の変形モデルを実時間で構築する。姿勢推定の中で得られる変形データに対して増分的な主成分分析を実行することで、変形モデルの逐次的な更新を行う。これにより、事前の学習を伴わずに頭部変形モデルを構築し、推定と並行してモデルの精度を随時向上させる機構を実現している。

第2の手法では、単眼カメラでの推定を実現するために、事前に構築した汎用的な顔形状モデルを用いる。顔形状の個人内変動（変形）と個人間変動（個人差）を分離して表現したパラメータ分離モデルの元で、二つの異なる処理を統合することで実時間頭部姿勢推定を実現する。一つはパーティクルフィルタを利用した姿勢、変形の時系列推定であり、もう一つはバンドル調整の枠組みを利用した個人差調整である。二つの手法を統合することで、各パラメータの性質の違いに対してそれぞれ適切なアプローチによる推定を実現することができる。これにより、不特定多数のユーザに対して事前準備や手動の初期化を伴うことなく、顔変形を含む3次元頭部姿勢推定を単眼カメラで実現する。

本論文では、精度・安定性の高い複眼推定手法と設置コストが低く柔軟な応用が可能な単眼推定手法により、多様な目的と設置状況への対応が可能な、真に実用に適した頭部姿勢推定技術を提案している。さらに、各々の手法に関する評価実験を行い、その結果についても報告する。

目次

第 1 章	序論	1
1.1	研究の背景	1
1.2	研究の目的	2
1.3	本論文の構成	3
第 2 章	関連研究	4
2.1	実時間頭部姿勢推定	4
2.2	変形モデルの複数人対応	6
2.2.1	個人毎の顔形状モデル作成	7
2.2.2	汎用的顔形状モデルの利用	8
2.3	本研究のアプローチ	10
第 3 章	顔変形モデルの自動構築を伴う頭部姿勢推定	12
3.1	はじめに	12
3.2	顔形状モデルの自動構築	13
3.2.1	顔形状モデル	14
3.2.2	形状モデルの初期化	15
3.2.3	頭部姿勢の推定結果を用いた特徴点位置の再計算	16
3.2.4	CCIPCA による顔形状モデルの自動構築	20
3.3	頭部変形モデルを用いた頭部姿勢推定	22
3.3.1	パーティクルフィルタを用いた頭部姿勢と顔形状の推定	22
3.3.2	Halfway Partitioned Sampling	25
3.4	評価実験	25
第 4 章	パラメータ分離モデルを用いた単眼による頭部姿勢推定	30
4.1	はじめに	30
4.2	頭部変形モデルの構築	32

4.2.1	モデル構築に用いるテンソル解析	32
4.2.2	N-mode SVD による形状パラメータの分離	36
4.3	バンドル調整の枠組みによる個人差パラメータ推定	40
4.3.1	バンドル調整の問題設定	41
4.3.2	バンドル調整の逐次実行	44
4.4	顔形状推定を伴う頭部姿勢推定	47
4.4.1	顔検出と顔形状・姿勢の自動初期化	47
4.4.2	パーティクルフィルタにおける多重線形モデルの利用	49
4.4.3	2次元特徴点座標の再計算	50
4.5	評価実験	52
第5章	結論	61
5.1	本研究のまとめ	61
5.2	今後の課題	62
	謝辞	65
	参考文献	66
	発表文献	71

目次

1.1	操作補助への応用	2
1.2	コミュニケーション支援への応用	2
2.1	Active Appearance Models [27]	5
2.2	メッシュモデルの例 [31]	5
2.3	特徴点モデルの例 [43]	5
2.4	因子分解法による非剛体の顔形状推定 [5]	8
2.5	パラメータ分離顔形状モデルの例 [8]	9
3.1	頭部姿勢推定システムの流れ	13
3.2	顔形状モデル構築の流れ	14
3.3	3次元顔形状モデルの概略	15
3.4	顔形状の初期化	16
3.5	特徴点の推定位置と正確な位置	17
3.6	パーティクルフィルタにおける推定の流れ	23
3.7	実験で利用したカメラ	26
3.8	頭部姿勢推定システムの概観	26
3.9	頭部姿勢推定の結果画像	28
3.10	頭部剛体/変形モデルを用いた頭部姿勢推定の結果	29
4.1	単眼システムの概要	31
4.2	3階のテンソルの展開	33
4.3	Singular Value Decomposition	34
4.4	モデル化の対象となる顔変形の例	36
4.5	モデル構築に用いるデータテンソル	37
4.6	パラメータ分離モデルにより生成される顔形状の例	39
4.7	バンドル調整の概要	41

4.8	姿勢推定の処理の流れ	47
4.9	顔モデルと頭部姿勢の初期化	48
4.10	頭部姿勢推定の結果画像（実験 A）	55
4.11	推定された顔特徴点座標（頭部座標系）の平均誤差（実験 A）	56
4.12	頭部姿勢推定の結果（実験 A）	57
4.13	頭部姿勢推定の結果（実験 B）	58
4.14	頭部姿勢推定の結果（実験 C）	60

表 目 次

4.1	頭部姿勢の推定誤差（実験 A）	54
4.2	頭部姿勢の推定誤差（実験 B）	56
4.3	頭部姿勢の推定誤差（実験 C）	59

第1章 序論

1.1 研究の背景

近年、高速なインターネット基盤の発達やデバイスの小型化、多様化に伴い、コンピュータとそれが構築するネットワークは様々な形で人間の日常生活の中に浸透しつつある。こうした中で、人とコンピュータの柔軟なインタラクションを実現するための新たなインタフェース技術への要求はますます高まっている。ビジネスやコミュニケーションのプラットフォームとしての情報空間の存在感がもはや無視できないほどに大きくなっている現在、人と情報空間とのインタラクション技術を見直すことは、特定のアプリケーションにおける機能改善以上の重要な意味を持ち得る。

現在のPCではデスクトップメタファにもとづく GUI (Graphical User Interface) が主流となっているが、これは2次元のディスプレイとマウス・キーボード、という入力デバイスに最適化されたインタフェースになっている部分がある。アプリケーションのレベルでは、現在も日々新たな UI の提案が行われているが、入力デバイスの制限から脱することは難しい。特にゲームやモバイル機器などにおいては、加速度センサやカメラなどを入力に用いるデバイスが増えてきているが、現在実用化されているセンサではまだまだ低レベルな情報しか取得できないこともあり、本質的なインタフェースの刷新には至っていない。高度な情報のセンシングを安価な装置で実現するための基礎技術が未だ不足していることが、新たなインタフェース技術の可能性を探る上でのボトルネックになっている側面も大きいと考えられる。こうした文脈において、高度なセンシング技術を現実的な応用が可能な形で実現させることへの期待は大きい。

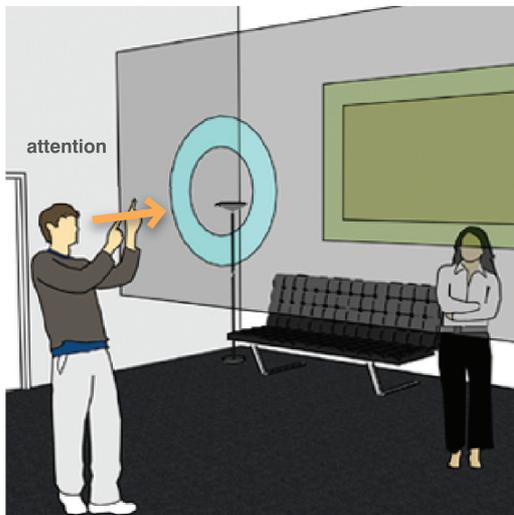


図 1.1: 操作補助への応用

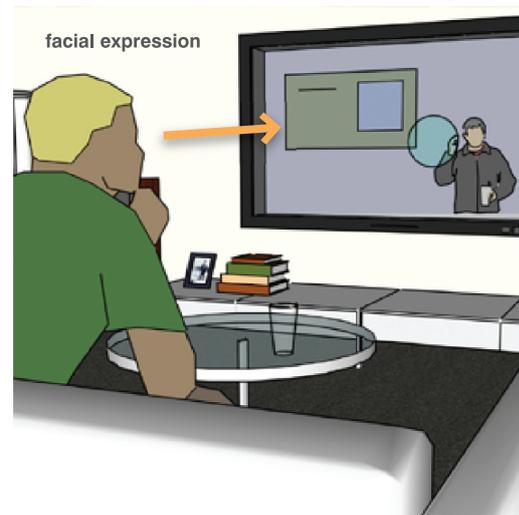


図 1.2: コミュニケーション支援への応用

1.2 研究の目的

本研究ではこうしたセンシング技術の中でも特に，ユーザの顔に関する情報に注目する．顔は人の意識や感情に関して多くの情報を有しているほか，ディスプレイなど多くのデバイスは人と対置されるため，デバイス自身に注目する人の顔は実用上観測しやすい対象でもある．したがって，図 1.1，図 1.2 に示すように，柔軟なインタラクションを考える上で応用の可能性が大きい．

本研究では特に，人の 3 次元頭部姿勢と顔変形を実時間で計測するためのシステム構築をその目的とする．現在，携帯電話やデジタルカメラなど，2 次元の顔検出とそれを応用した技術は既に多くの場面で実用化されてきている．一方，3 次元の頭部姿勢や詳細な表情認識に関しては未だ決定的な技術は存在せず，現在も活発な研究・開発が続けられている．

実用性という観点から見ると，いかにユーザに負担をかけないセンシングを実現するかが重要になる．頭部姿勢や変形を高精度で測定するための一つのアプローチとして，ユーザの側で磁気センサや特殊なマーカを装着するという手法が考えられる．こうした手法は特殊なアプリケーションの操作を目的とした場合は有効だが，日常生活の中で利用する場合を考えると現実的ではない．また，デバイス側で赤外線カメラや 3D スキャナなどの特殊なセンサを利用するアプローチも考えられるが，こうした手法は実現コストが高くなり，また応用の範囲が限定される

という欠点がある。

そこで本研究では、通常のカメラ画像を元に3次元頭部姿勢と顔変形を計測する手法を提案する。真に実用に適した3次元頭部姿勢推定システムを構築するために、特に次の2点に主眼を置く。

- 不特定多数のユーザに対して自動的に推定を開始できる
- 発話や表情変化に伴う顔変形が起こった場合でも、安定して推定を継続できる

この2点を満たす頭部姿勢推定システムとして、二つの異なるアプローチに基づく手法を提案する。

1.3 本論文の構成

不特定多数のユーザを対象として変形を含む姿勢推定を行う際には、個人ごとに異なる変形を捉えるためのモデルをいかに用意するかが重要になる。

第3章で述べる第1の手法では、複数台のカメラを用いてユーザ頭部の3次元形状を直接計測することでこれを実現する。一般的に、変形モデルを構築するためには長い時間をかけて取得した大量のデータに対して統計的な解析を行う必要がある。それに対し本手法では、変形データの取得及びモデル構築を姿勢推定と並行して実行することにより、事前の準備無しにユーザ独自の変形モデルを獲得し、それを利用した変形推定を可能にしている。

一方、第4章で述べる二つめの手法では単眼カメラでの推定を実現するために、事前に構築した顔形状モデルを利用する。複数人のデータを含む、情報量の多い事前知識を実時間システムの中で効率よく扱うために、顔形状の個人内変動と個人間変動を分離して記述したモデルを構築する。このようなモデルに対して、複数フレームの観測に基づく個人差の調整を実時間推定と並行して実行する。

以下、まず第2章では関連研究についてまとめる。次に第3章と第4章で個々の手法の詳細について述べ、最後に第5章において各々の特徴を整理し、今後の課題や応用について述べる。

第2章 関連研究

2.1 実時間頭部姿勢推定

まず本節では、コンピュータビジョンを利用した実時間頭部姿勢推定に関する関連研究について紹介する。ここでは、顔のモデル化手法と推定手法の二つの観点から関連研究を整理する。

モデル化手法の観点から見ると、頭部姿勢推定手法は大きく二つに分類することができる。顔全体の見えを2次元の画像パターンとして捉えるアピアランスベースの手法と、3次元の詳細な顔形状モデルを利用するモデルベースの手法である。さらに、モデルベースの手法は利用するモデルの種類に応じてメッシュモデルに基づく手法と特徴点モデルに基づく手法に分けて考えることができる。本節ではそれぞれのアプローチに関して、代表的な例及びその長所・短所を整理するとともに、本研究のベースとなる先行研究の位置付けを整理する。

アピアランスベースの形状モデルの代表的な例としては、Cootesらによって提案されたAAM (Active Appearance Models) [6] が挙げられる。AAMは、顔領域のテクスチャと形状をそれぞれ図2.1のような線形基底で表現したモデルになっている。MatthewsらはAAMを利用して顔向きによる見えの違いをモデル化し、実時間の頭部姿勢推定を実現している[27]。また、顔を空間内の2次元平面として扱う手法[17]や楕円として扱う手法[4]、円柱として扱う手法[20]など、顔全体の見えのモデルを3次元空間内で利用する手法も古くから数多く提案されている。こうしたアピアランスベースの手法は画像中の細かい差異に対して比較的安定であるなどのメリットがあるが、顔形状の厳密なモデル化は困難であり、詳細な頭部姿勢の推定を行うのには向いていない。

これに対して、顔の3次元形状モデルに基づく推定を行うモデルベースの手法として、まずは図2.2のような顔全体のメッシュモデルを利用する手法が挙げられる[12, 13, 36, 31]。Xiaoらは3次元メッシュモデルをAAMに対する拘束条件として

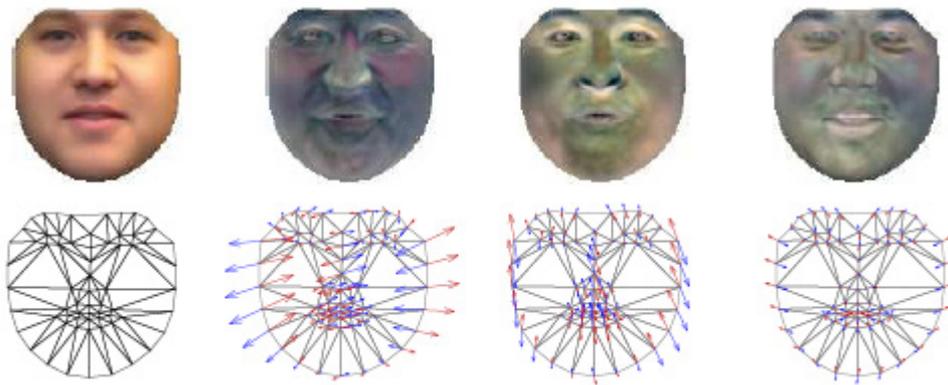


図 2.1: Active Appearance Models [27]



図 2.2: メッシュモデルの例 [31]



図 2.3: 特徴点モデルの例 [43]

利用することで、より厳密な3次元頭部姿勢と変形を推定する手法を提案している[36]。また、Vacchettiらの手法[31]では、バンドル調整の枠組みを利用してこのような3次元メッシュモデルを用いた剛体オブジェクトの追跡を実現している。

一方、よりシンプルな3次元顔形状モデルとして、図2.3のように3次元の特徴点集合として表現されたモデルを用いる手法も多い[14, 15, 26, 28, 42, 43]。Geeらの手法[14]では、目や鼻など顔の各器官の3次元位置関係を手作業で測定することで得られる顔形状モデルを用いて、RANSAC (RANDOM SAMPLING CONSENSUS) に基づく姿勢推定を行っている。Matsumotoらの手法[26]では、ステレオカメラにより得られる3次元特徴点モデルを利用して、入力画像に対するモデルのフィッティングによる姿勢推定を実現している。

こうした顔の3次元形状モデルに基づく手法は、より詳細な頭部姿勢推定を行うことが可能となる。前者のメッシュモデルは最も詳細な顔形状の表現が可能であり、感情認識など高度な応用を考える上でも有用であるが、その分モデル自体の情報量も多く、実時間システムで扱うには計算コストや安定性の面での工夫が必要になる。それに対して後者の特徴点モデルは必要最低限の情報量で顔形状を表現でき、また計算上も収束しやすく安定した推定が可能になるというメリットがある。

本研究のベースとなる岡らの手法[43]では、こうした特徴点ベースの顔形状モデルを用いた頭部姿勢推定を提案している。初期化時に自動構築するユーザ独自の3次元剛体顔形状モデルを用いて、パーティクルフィルタを用いた頭部姿勢推定を行う。これにより、部分遮蔽や照明変動、そしてユーザの突発的な動作に対しても頑健なシステム構築を可能にしている。

2.2 変形モデルの複数人対応

前節で紹介した手法の中には、人間の顔を剛体とみなしたモデルを利用して頭部姿勢を推定するものが見られる[14, 26, 31, 43]。しかし、人間の顔は発話や表情変化などのために変形する場合があるため、それを剛体とみなすのはあまり現実的な仮定ではない。実際、頭部剛体モデルを利用した手法では、顔の変形が起こった場合に精度の劣化や追跡の失敗が生じることが大きな問題となっている。

この問題を解決するためには、顔形状の変化を表現することができる変形モデ

ルを用いる必要がある。本節では，こうした顔形状の変形モデルの利用に関して，必ずしも実時間のシステム構築を目的としていない顔形状推定技術も含めて関連研究を紹介する。

不特定多数のユーザを対象として変形モデルを利用するためには，大きく分けて次の二つのアプローチが考えられる。

- ユーザ毎にそれぞれ独自の変形モデルを作成する
- 個人間の差異も含んだ汎用的な顔形状モデルを利用する

本節ではこの二つの観点から関連研究を整理する。

2.2.1 個人毎の顔形状モデル作成

変形モデルを作成するための手法としてもっとも一般的なのは，ユーザ毎に事前の学習により専用の変形モデルを作成する手法 [3, 15, 27, 28, 42] であると言える。

前述の Matthews らの手法 [27] では，実時間の姿勢推定を行う際，対象となるユーザに対する専用のモデルを利用している。Gokturk らの手法 [15] では，事前にステレオカメラを用いた表情変化の追跡を行い，獲得されたデータに対して主成分分析を行うことで変形モデルを作成する。これを利用することで，単眼のカメラ入力に対して変形を含む姿勢推定を実現している。Zhu らの手法 [42] では同様の3次元変形モデルを元に，2次元の特徴点軌跡から姿勢と変形を実時間で復元する手法を提案している。こうした手法は，モデルの精度に関しては最も理想的な状態を得られるが，実際の応用を考える上で，ユーザ毎に長い時間をかけて事前に変形モデルを準備するのは現実的ではない。

これを解決するための手法としては，入力画像に対して姿勢・変形の推定と顔形状のモデリングを同時に行うことが考えられる。これに関する一つのアプローチとして，Structure from Motion の枠組みを変形を含む対象に適用することで事前の準備無しに顔形状の変形を推定する手法が提案されている。Bregler らは Tomasi-Kanade の因子分解法を非剛体の形状推定に拡張する手法を提案しており [5]，2次元の追跡結果からの形状復元を変形に関する事前の知識無しで実現している。この非剛体因子分解法はその後も様々な拡張がなされており，顔形状推定を目的とした研究例も報告されている [9, 10, 37]。しかし，3次元モデルを利用した実時間頭部姿勢推定においてこうした手法を応用している例は無い。

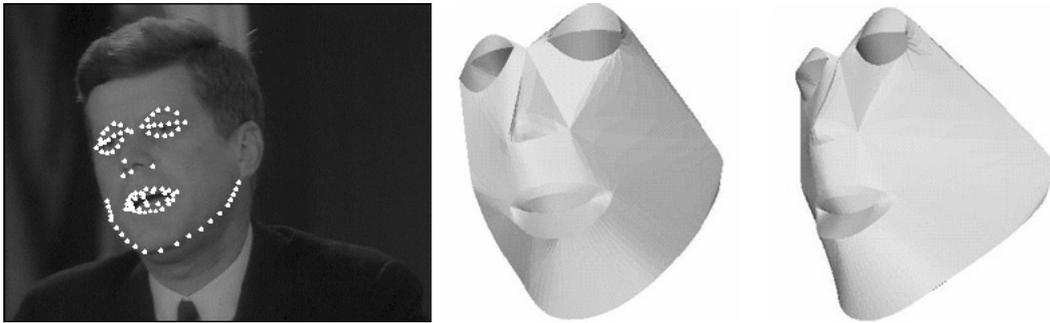


図 2.4: 因子分解法による非剛体の顔形状推定 [5]

2.2.2 汎用的顔形状モデルの利用

一方，事前に複数人の変形データを含む顔形状モデルを用意する，というアプローチにもいくつかの研究例が存在する．

その一つの方法として，個人差に基づく変動も同じ変形と捉え，単一の形状パラメータで全ての変動を表現するモデルを作成する手法がある．AAM に関しては，Matthews らの文献 [27] でも学習対象を複数人のデータに拡大した Generic AAM に関する議論が為されており，Gross らはこの Generic AAM に関してフィッティング性能を改善するための新たなアルゴリズムを提案している [16]．しかし，計算コストの高さや収束の難しさはこのような単一パラメータのモデルが本質的に持つ問題であり，実時間システムにおいて安定した推定を実現するには困難が伴う．

そこで，汎用的な顔形状モデルを作成する上で必要となる次の二つの要素をそれぞれ個別のパラメータで表現するモデルを用いた手法がいくつか提案されている．

- 眉の上下や口の開閉といった，個人内の変形に対応する要素
- 目の位置や鼻の高さといった，個人差に対応する要素

Dornaika らの手法 [12, 13] では，Candide [2] と呼ばれる 3 次元メッシュモデルが用いられている．Candide はもともと 3DCG アニメーションのために作成されたモデルであり，顔変形は規格として定められた基底の線形和として表現され，変形を表す Action Units と個人差を表す Shape Units が明示的に分離されている．文献 [12] ではパーティクルフィルタの枠組みを応用した単眼姿勢推定を，さらに文献 [13] ではステレオカメラを併用した追跡の安定化手法を提案している．このとき，追跡を行う上で対象となるユーザに関して変動し得るのは変形を表すパラメータ

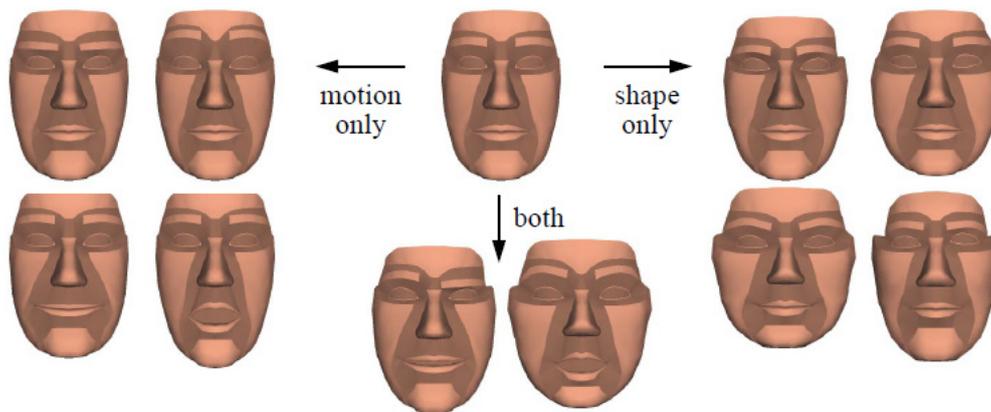


図 2.5: パラメータ分離顔形状モデルの例 [8]

のみであり，Dornaikaらはユーザ毎に個人差パラメータは固定として変形の推定を行っている．しかし，個人差パラメータの具体的な値に関してはユーザごと事前に決定する必要があり，また追跡の中でこれを調整する仕組みも備えていない．

DeCarloらの手法[7, 8]でも同様に，手作業でパラメータの分離が行われた図 2.5のような3次元メッシュモデルを利用している．文献[7]では，画像のエッジ情報から個人差も含めたパラメータ全体を，そしてオプティカルフローの情報からは変形パラメータのみを調整する手法を提案している．さらに文献[8]では上記の手法にオプティカルフローの推定残差を最小化するパラメータ調整を組み合わせ，より精度の高い推定を実現している．彼らはこの二つのパラメータの振舞いの違いをより高度な形で利用しているが，この手法は計算コストが高く，実時間の推定システムには応用されていない．

一方，VlasicらはN-mode SVDに基づく多重線形顔形状モデルを提案している[34]．上記のような手作業でパラメータを分離したモデルとは異なり，より統計的な性質に基づくパラメータの分離が可能になる．形状推定を行う際には，彼らもパラメータの性質の違いに注目しており，個人差に相当する，同一ユーザに対して一定の値を取るパラメータに関しては，入力画像列全体に対してパラメータの最適化を行っている．しかし，この計算は撮影した動画の後処理を前提としているため，実時間の処理にそのまま適用することはできない．

2.3 本研究のアプローチ

これまでに紹介したように，不特定多数のユーザを対象とする実時間頭部姿勢推定システムにおいて，変形を含む顔形状モデルを扱うのは現在でも難しい課題となっている．それに対し本研究では，特徴点ベースの顔形状モデルを用いた頭部姿勢推定システム [43] を元に，任意のユーザに対して顔変形を含む姿勢推定を実現する手法を提案する．本研究で述べる二つの手法に関して，それぞれ従来手法に対する位置づけを整理する．

顔変形モデルの自動構築を伴う頭部姿勢推定

まず，第3章で述べる手法では，2.2.1節で述べた研究例と同様に，追跡対象となるユーザに対して独自の变形モデルを作成するアプローチを取る．

しかし，本研究ではこれらの例とは異なり，ステレオカメラを用いた追跡の中で得られるユーザの顔形状情報を利用して，追跡と並行して増分的に変形モデルを構築する．毎フレームごとにモデルの更新を行い，次フレームの姿勢推定には更新されたモデルを利用するため，顔変形と姿勢の推定精度を逐次的に向上させることができる．したがって，従来手法のように変形を推定するためのユーザ毎の事前準備は必要なく，また非剛体 Structure from Motion 手法では実現できない実時間のシステム構築，および逐次的なモデル更新の追跡へのフィードバックを実現している．

パラメータ分離モデルを用いた単眼カメラによる頭部姿勢推定

第4章で述べる手法では，2.2.2節で述べた研究例と同様に，事前に作成した汎用的な顔形状モデルを利用する．

本研究では，顔形状の個人内変動（変形パラメータ）と個人間変動（個人差パラメータ）を分離して表現したモデルを用いる．しかし，モデルに対するパラメータ推定手法は従来手法とは異なり，時間依存の姿勢・変形パラメータ推定にはパーティクルフィルタを，時間非依存の個人差パラメータを含めた全体の推定にはバンドル調整を利用する．3次元モデル空間内での頑健な時系列推定と複数フレームの

観測を利用した個人差の調整をパラメータ分離モデルの元で統合することにより、
実時間システムとして、安定かつ精度の高い姿勢・変形推定を可能にしている。

第3章 顔変形モデルの自動構築を伴う頭部姿勢推定

3.1 はじめに

本章では、頭部変形モデルの構築と頭部姿勢の実時間推定を同時に実行するための手法を提案する。

本手法では、事前の準備なしに自動構築された頭部剛体モデルを用いて、頭部姿勢の実時間推定を開始する。それと同時に、姿勢推定の結果を手がかりとして頭部の各特徴点の正確な動きを実時間で計測する。こうして計測した特徴点の動きを分析することにより、姿勢推定を継続しながらも、新たな頭部変形モデルを構築する。さらに、更新された変形モデルを次フレームからの推定に利用することで、姿勢推定とそれに基づく特徴点の再計測の精度を逐次的に向上させることが可能になる。

本研究の主要な貢献としては次の3点が挙げられる。

1. 特別な事前準備なしに顔変形を伴う頭部姿勢運動を実時間で追跡する。
2. 頭部変形モデルを連続的に更新する。
3. 頭部姿勢と顔形状の推定性能を逐次的に向上させる。

本手法で構築した頭部変形モデルによる頭部姿勢推定性能の向上について評価実験を通して明らかにする。

本章で述べる頭部姿勢推定システムの流れは図3.1のようになる。ユーザの顔形状モデルを構築する *Model construction step* と、入力画像フレームから頭部姿勢と顔変形を推定する *Pose estimation step* により構成されている。

本節以降、第3.2節と第3.3節では、それぞれ *Model construction step* と *Pose estimation step* について説明する。最後に第3.4節で本手法に関する実験結果を示す。

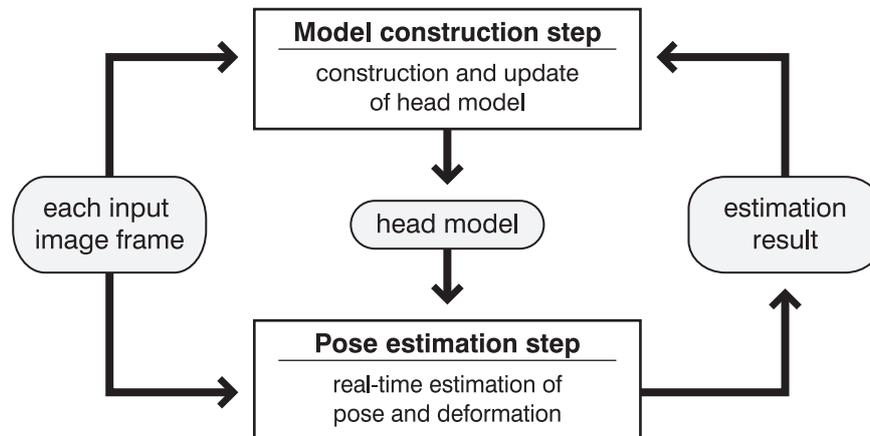


図 3.1: 頭部姿勢推定システムの流れ

なお，本論文では校正済みのカメラを2台用いる場合について説明するが，この台数は理論的な拡張なしに増設することが可能である．

3.2 顔形状モデルの自動構築

本節では，ユーザの顔形状モデルを獲得するための手法について説明する．これは図3.1の *Model construction step* での処理に対応する．

さらに詳細なモデル構築の流れを，図3.2に示す．まず，初期化部 (*Initialization step*) では暫定的に剛体の顔形状モデルを自動的に構築する．初期化が完了すると，次の3つの段階を繰り返すことで連続的なモデルの更新を行う．最初の段階 (*Pose estimation step*) では，入力画像から実時間で頭部姿勢 p_t と顔変形 a_t を推定する．この推定手法の詳細に関しては第3.3節で述べる．第2段階 (*Feature-point recalculation step*) では，第1段階で推定された p_t と a_t を元に正確な特徴点位置を再計算する．こうして新たに得られた特徴点座標を利用して，最後に第3段階 (*Deformation analysis step*) で主成分分析による顔形状モデルの更新を行う．

これらの処理を経て更新された顔形状モデルは次フレームでの頭部姿勢推定，すなわち，*Pose estimation step* で使用する．これにより，顔形状モデルを随時更新するとともに，頭部姿勢と顔変形の推定性能を逐次的に向上させるような機構を実現する．

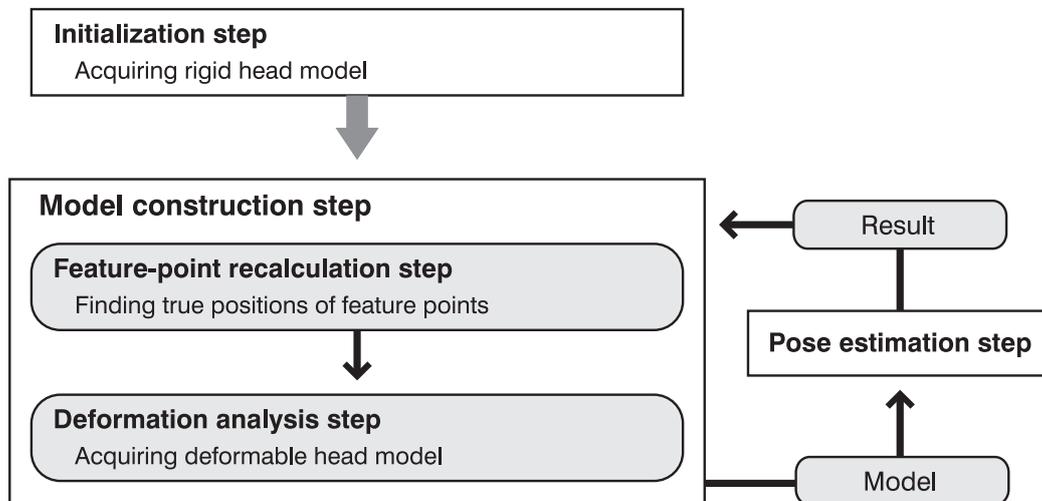


図 3.2: 顔形状モデル構築の流れ

3.2.1 顔形状モデル

本手法では、図 3.3 に示すような K 個の特徴点で構成される顔形状モデルを用いる。本論文では $K = 10$ で固定されており、各点は両目の両端と口の両端、両鼻孔、そして両眉の内側の端点に対応する（図 3.4 における “+” マーク）。各々の特徴点は 2 つの要素により構成されている。1 つはフレーム t における特徴点の 3 次元座標であり、ユーザ頭部に固定されたモデル座標系内での座標として表現される。もう 1 つは特徴点のテンプレート画像であり、左右 2 台のカメラに対して 1 枚ずつ保有している。以下、 M_t を K 個の特徴点の 3 次元座標で構成された $3K$ 次元形状ベクトルとする。また、 T_L は左カメラに対する K 個のテンプレートの集合を表すものとし、同様に T_R を右カメラに対するテンプレート集合とする。本論文における頭部姿勢は、頭部姿勢はカメラ位置を基準とした世界座標系からモデル座標系への並進及び回転として定義される。

形状ベクトル M_t は、次のように基底形状ベクトルの線形和で表現される。

$$M_t = \bar{M} + \mathcal{M}a_t \quad (3.1)$$

このとき、 \bar{M} は平均形状ベクトル、 \mathcal{M} は B 個の基底形状ベクトルを並べた $3K \times B$ の基底形状行列、 a_t は \mathcal{M} の B 次元係数ベクトルである。すなわち、 a_t の各要素が顔変形のパラメータとなる。これにより少数のパラメータで顔形状の変化を表現することを可能とするとともに、 B の大きさを制限することで頭部変形モデル

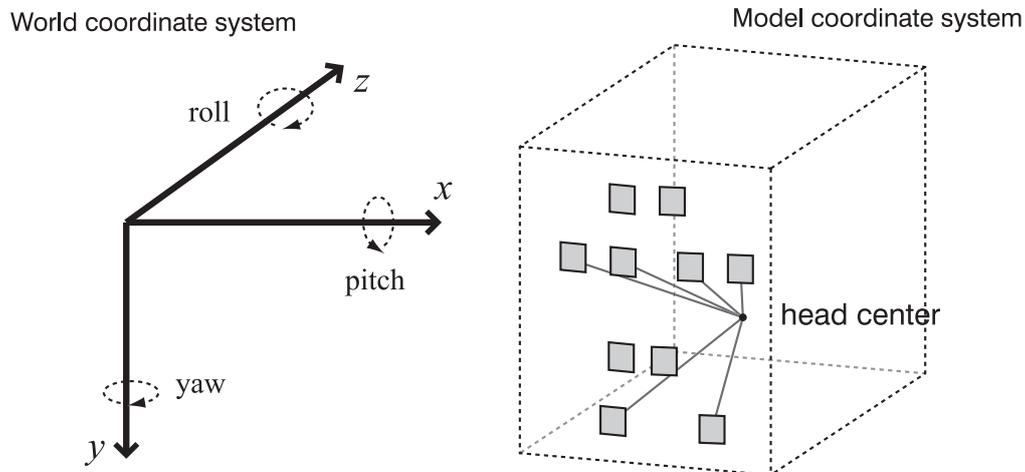


図 3.3: 3次元顔形状モデルの概略

が不自然に変形するのを抑制する効果も与える．本論文では B は経験的に 5 で固定しており，本手法で用いる顔モデルに対しては十分な累積寄与率が得られることを確認している．なお， \bar{M} と \mathcal{M} を逐次的に計算する方法については 3.2.4 節で説明する．

3.2.2 形状モデルの初期化

まず，本節で初期化部 (*Initialization step*) について説明する．初期化部では，暫定的な顔形状モデルとして，ユーザ頭部の 3次元剛体モデルを自動的かつ高速に構築する．ここでは，オムロン社で開発された OKAO ビジョンライブラリを利用することで，入力画像から自動的に顔および 6 個の顔特徴点（図 3.4 中，赤で示した特徴点に対応）を検出する [21]．さらに残りの $(K - 6)$ 個の特徴点（図 3.4 中，緑の点に対応）については，既知の特徴点位置に対して事前知識に基づき定めた探索範囲の中から，追跡に適した画像特徴を持つ点を検出する [30]．具体的には，探索範囲内の各画素に対して，式 (3.2) に示すような行列 D を計算する．

$$D = \begin{pmatrix} \sum_{\text{ROI}} \frac{\partial I^2}{\partial x} & \sum_{\text{ROI}} \frac{\partial I}{\partial x} \frac{\partial I}{\partial y} \\ \sum_{\text{ROI}} \frac{\partial I}{\partial x} \frac{\partial I}{\partial y} & \sum_{\text{ROI}} \frac{\partial I^2}{\partial y} \end{pmatrix} \quad (3.2)$$



図 3.4: 顔形状の初期化

ここで、 D の各要素はその画素を中心とした窓領域 (Regions Of Interest ; ROIs) における画素値 I の微分係数の和を示す。この行列 D の最小固有値が設定した閾値を超えている点の中から、他の特徴点との距離が十分離れている点を追跡のための特徴点として採用する。

まず、以上の処理を基準となる左カメラからの入力画像に適用して K 個の特徴点を検出する。次に、これらの特徴点に対応する点 (図 3.4 中、黄色で示した点) を右カメラからの入力画像中で探索するために、エピポーラ線上でテンプレートマッチングにもとづく探索を行う。こうして得られた 2 次元の特徴点位置座標を元に、各特徴点の 3 次元位置座標を計算する。そして、頭部に固定されたモデル座標系を定義するとともに、モデル座標系内での顔の 3 次元形状 M を決定する。最後に、この M とテンプレート集合 T_L, T_R を頭部モデルとして登録する。以上の処理により頭部モデルの登録が成功した場合のみ、これ以降に述べる変形モデルの逐次更新及び頭部姿勢の追跡処理に移り、失敗した場合は登録に成功するまでこれら一連の処理を繰り返す。なお、初期化直後は剛体モデルしか存在しないため、変形パラメータ a_t は 0 に設定する。

3.2.3 頭部姿勢の推定結果を用いた特徴点位置の再計算

本節では、図 3.2 における *Feature-point recalculation step* について説明する。

正確な頭部変形モデルを構築するためには、各特徴点の正確な 3 次元座標が必要になる。後に詳細を述べる *Pose estimation step* では、式 (3.1) を元にフレーム

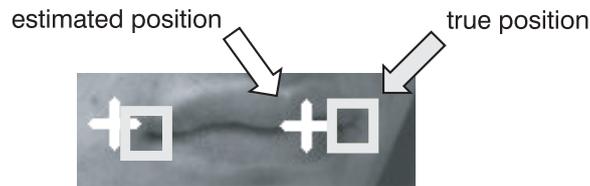


図 3.5: 特徴点の推定位置と正確な位置

ごとの顔形状 M_t を推定する。しかし、本システムでは逐次的なモデル更新を行うため、未知の変形が起こった場合には必ずしも推定された形状が正確な特徴点位置と一致するとは限らない(図 3.5)。

そこで本手法では、*Pose estimation step* で推定した頭部姿勢 p_t と顔形状 M_t を元に、改めて各特徴点の正確な位置を再計算する。ここで、モデル座標系における K 個の特徴点の正確な 3 次元位置座標を表す $3K$ 次元ベクトルを M'_t とし、以下、 M'_t を真形状と呼ぶ。

本手法では、 M'_t についての誤差関数 E_t を定義し、 E_t を最小化することによって M'_t を決定する。

$$E_t = E_t^I + \epsilon E_t^M \quad (3.3)$$

ここで、 E_t^I は特徴点周辺の見えに関する誤差を表し、 E_t^M は推定形状 M_t に対する幾何学的な誤差を表している。このとき、 ϵ は定数であり、本手法では経験的に 2000 に固定している。以下、 E_t^I と E_t^M について具体的に説明する。

E_t^I を定義する前に、まず投影関数 \mathcal{P}_h を定義する。これは、頭部姿勢 p_t に応じた並進や回転を真形状 M'_t に適用した上で、各特徴点を画像平面 h に投影する関数である。

$$\mathbf{m}_{h,t} = \mathcal{P}_h(p_t, M'_t) \quad (3.4)$$

$\mathbf{m}_{h,t}$ は K 個の投影点の 2 次元座標で形成される $2K$ 次元ベクトルである。

これを用いて、特徴点周辺の見えに関する誤差 E_t^I を次のように定義する。

$$E_t^I = \sum_{\substack{\text{ROI} \\ h \in \{L, R\}}} \left\{ \rho \left\| \mathbf{I}_t^h(\mathbf{m}_{h,t}) - \mathbf{I}_{t-1}^h(\mathbf{m}_{h,t-1}) \right\|^2 + \left\| \mathbf{I}_t^h(\mathbf{m}_{h,t}) - \mathbf{I}_1^h(\mathbf{m}_{h,1}) \right\|^2 \right\} \quad (3.5)$$

ここで、 $\mathbf{I}_t^h(\mathbf{m}_{h,t})$ は K 次元輝度ベクトルを示す。 $\mathbf{I}_t^h(\mathbf{m}_{h,t})$ の k 番目の要素は、入力画像 I_t^h における $\mathbf{m}_{h,t}$ の k 番目の 2 次元座標での輝度を表す。式 (3.5) の第 1 項

は、現在の画像 I_t^h 中の K 個の注目領域と直前の画像 I_{t-1}^h 中の注目領域との見えの誤差を表す標準的な誤差関数である．一方、第2項は現在の画像 I_t^h 中の注目領域と追跡開始時の画像 I_1^h 中の注目領域との見えの誤差を表す．この項は Gokturk らの手法 [15] でも使用されており、特徴点のドリフトを回避する効果が報告されている．また、 ρ は第1項と第2項の間の比率を表す．現在は経験的に、 ρ を4に、注目領域の大きさを 16×16 画素に設定している．

E_t のもう一方の項 E_t^M は次のように、推定形状 M_t と真形状 M_t' との誤差として定義される．

$$E_t^M = \|M_t' - M_t\|^2 \quad (3.6)$$

この項の導入により、ユーザの頭部姿勢が大きく変化した場合であっても、特徴点を見失うことなく追跡できるようになる．頭部変形モデルの更新により M_t が正確な形状に近づくにつれて、より安定して M_t' を計算することが可能となる．

以上の定義により得られる E_t を最小化することにより、 M_t' を算出する．具体的には、次のような手順で前フレームとの差分形状 $dM = M_t' - M_{t-1}'$ を計算する [15]．

まず、輝度ベクトル $I_t^h(m_{h,t})$ を M_{t-1}' の近傍で1次の項までテイラー展開することで次のような近似式を得る．

$$\begin{aligned} I_t^h(m_{h,t}) &= I_t^h(\mathcal{P}_h(\mathbf{p}_t, M_t')) \\ &= I_t^h(\mathcal{P}_h(\mathbf{p}_t, M_{t-1}' + dM)) \\ &= I_t^h(\mathcal{P}_h(\mathbf{p}_t, M_{t-1}')) + \hat{K}_t^h dM \end{aligned} \quad (3.7)$$

ただし、

$$\hat{K}_t^h = \frac{\partial I_t^h}{\partial \mathbf{m}_{h,t}} \bigg|_{\hat{\mathbf{m}}_{h,t}} \frac{\partial \mathcal{P}_h}{\partial M_t'} \bigg|_{\mathbf{p}_t, M_{t-1}'} \quad (3.8)$$

$$\hat{\mathbf{m}}_{h,t} = \mathcal{P}_h(\mathbf{p}_t, M_{t-1}') \quad (3.9)$$

とする．これを式 (3.5) に代入することにより、 E_t^I は次のように dM の関数として表現できる．

$$E_t^I = \sum_{\substack{\text{ROI} \\ h \in \{L, R\}}} \left\{ \rho \| \hat{K}_t^h dM + \Delta I \|^2 + \| \hat{K}_t^h dM + \Delta I_0 \|^2 \right\} \quad (3.10)$$

ただし

$$\Delta \mathbf{I} = \mathbf{I}_t^h(\hat{\mathbf{m}}_{h,t}) - \mathbf{I}_{t-1}^h(\mathbf{m}_{h,t-1}) \quad (3.11)$$

$$\Delta \mathbf{I}_0 = \mathbf{I}_t^h(\hat{\mathbf{m}}_{h,t}) - \mathbf{I}_1^h(\mathbf{m}_{h,1}) \quad (3.12)$$

とする．

また， E_t^M に関しては，式 (3.6) から次のように dM の関数として記述できる．

$$E_t^M = \|dM + M'_{t-1} - M_t\|^2 \quad (3.13)$$

したがって，式 (3.10) と式 (3.13) から E_t は dM に関する 2 次関数となり，次式を解くことで E_t を最小にする dM が得られる．

$$\frac{\partial E_t}{\partial dM} = 0 \quad (3.14)$$

E_t^I ， E_t^M の各々を dM で偏微分した結果は次のようになる．

$$\frac{\partial E_t^I}{\partial dM} = \sum_{\substack{\text{ROI} \\ h \in \{L, R\}}} \left\{ 2(\rho + 1) \hat{\mathbf{K}}_t^{hT} \hat{\mathbf{K}}_t^h \right\} dM + \sum_{\substack{\text{ROI} \\ h \in \{L, R\}}} \left\{ 2 \hat{\mathbf{K}}_t^{hT} (\rho \Delta \mathbf{I} + \Delta \mathbf{I}_0) \right\} \quad (3.15)$$

$$\frac{\partial E_t^M}{\partial dM} = 2dM + 2(M'_{t-1} - M_t) \quad (3.16)$$

したがって，式 (3.3)，式 (3.14)，式 (3.15)，式 (3.16) から dM は次式によって求められる．

$$dM = -D^{-1}d \quad (3.17)$$

ただし， D ， d は次のように定義する．

$$D = \sum_{\substack{\text{ROI} \\ h \in \{L, R\}}} \left\{ 2(\rho + 1) \hat{\mathbf{K}}_t^{hT} \hat{\mathbf{K}}_t^h \right\} + \epsilon \mathbf{E}_{3K \times 3K} \quad (3.18)$$

$$d = \sum_{\substack{\text{ROI} \\ h \in \{L, R\}}} \left\{ 2 \hat{\mathbf{K}}_t^{hT} (\rho \Delta \mathbf{I} + \Delta \mathbf{I}_0) \right\} + \epsilon (M'_{t-1} - M_t) \quad (3.19)$$

ここで $\mathbf{E}_{3K \times 3K}$ は $3K \times 3K$ の単位行列を示す．収束するまで式 (3.17) による dM の算出を繰り返すことで，最終的な真形状 M'_t を得る．

多くの場合は，この最小化処理により真形状 M'_t を正しく決定することができる．その一方で，*Pose estimation step* における頭部姿勢の推定誤差などの影響に

より, M'_t の中に剛体運動に対応する成分が含まれる場合も存在する. この成分を除去せずに頭部変形モデル構築し, その頭部モデルを用いて頭部姿勢を推定した場合, 頭部剛体運動と顔変形を適切に分離することが不可能になる. そこで, M'_t に含まれる剛体運動成分を除去するために, 次のような処理を行う [27]. まず, M'_1 から M'_{t-1} までの平均形状 \bar{M}'_{t-1} を計算する. その後, M'_t と \bar{M}'_{t-1} の対応する点同士の距離の2乗和が最小になるように, M'_t に3次元的な並進と回転を適用し, その結果を改めて M'_t として定義する. 以上の方法により, M'_t に剛体運動成分が含まれる場合には, その大部分を除去することが可能である. しかしながら, M'_t が正しく計算されている場合には悪影響を及ぼす可能性もある. そこで本手法では, M'_t と M_t との距離が一定の閾値を超えた場合のみ, 以上の処理を適用している.

3.2.4 CCIPCA による顔形状モデルの自動構築

本節では, 図3.2の *Deformation analysis step* について説明する.

ここでは, フレーム t までに得られた真形状 M'_1, M'_2, \dots, M'_t を元に平均形状ベクトル \bar{M}_t と基底形状行列 \mathcal{M}_t を計算する. 通常のPCA (主成分分析: Principal Components Analysis) では, データ全体の分散共分散行列について解析を行う必要があるため, 実時間システムの中で得られるデータに対して逐次的に基底行列を計算することはできない. それに対し本手法では, Candid Covariance-free Incremental PCA (CCIPCA) [35] と呼ばれる手法を利用することで, 個々の入力データが得られるごとに基底行列を更新することを可能にしている. 以下, 具体的な処理について簡単に説明する.

まず, 平均 \bar{M}_t は次のように増分的な推定が可能である.

$$\bar{M}_t = \frac{t-1}{t}\bar{M}_{t-1} + \frac{1}{t}M'_t \quad (3.20)$$

ここで, $\hat{M}_i = M'_i - \bar{M}_t$ とおく. この正規化された形状ベクトル \hat{M} の分散共分散行列 $S = E\{\hat{M}\hat{M}^T\}$ を知ることができれば, 基底形状ベクトル x は行列 S の固有ベクトルとして算出することができる. すなわち, 対応する固有値を λ とすると, x と λ は次式を満たす.

$$\lambda x = Sx \quad (3.21)$$

しかし，フレーム t までに得られたサンプルから母集団の分散 S を推定することはできない．そこで，式 (3.21) において S をサンプルの分散共分散行列に， x をフレーム i での推定値 x_i に置き換えることで，次のようなベクトル $v = \lambda x$ の近似式を得る．

$$v_t = \frac{1}{t} \sum_{i=1}^t \hat{M}_i \hat{M}_i^T x_i \quad (3.22)$$

さらに，右辺の x_i を 1 フレーム前の推定値で置き換えることで，次のように増分的な計算式が定義できる．

$$v_t = \frac{1}{t} \sum_{i=1}^t \hat{M}_i \hat{M}_i^T \frac{v_{i-1}}{\|v_{i-1}\|} \quad (3.23)$$

$$= \frac{t-1}{t} v_{t-1} + \frac{1}{t} \hat{M}_t \hat{M}_t^T \frac{v_{t-1}}{\|v_{t-1}\|} \quad (3.24)$$

ただし， $v_0 = \hat{M}_1$ とする．最大の固有値を $\lambda_{(1)}$ ，それに対応する第 1 主成分を $x_{(1)}$ とすると， v_t は $t \rightarrow \infty$ のとき $\pm \lambda_{(1)} x_{(1)}$ に収束する [40]．

一方，式 (3.24) からは第 1 主成分しか算出できないため，さらに次のような手順で第 2 主成分以降を算出する．第 j 主成分の算出に用いるサンプルを $\hat{M}_{i,(j)}$ とすると， $v_{i,(j)}$ を求めた後，次式により新たに $\hat{M}_{i,(j+1)}$ を計算する．

$$\hat{M}_{i,(j+1)} = \hat{M}_{i,(j)} - \hat{M}_{i,(j)}^T \frac{v_{i,(j)}}{\|v_{i,(j)}\|} \frac{v_{i,(j)}}{\|v_{i,(j)}\|} \quad (3.25)$$

これにより得られるサンプル $\hat{M}_{i,(j+1)}$ は $v_{i,(j)}$ の補空間におかれるため，これを式 (3.24) の入力として用いることで $v_{i,(j)}$ に直交するような $v_{i,(j+1)}$ を算出することができる．

つまり，新たなサンプル形状 M'_t が得られるごとに，まず式 (3.20) により平均を更新する．さらに， B 個の基底形状ベクトル全てに対して式 (3.24) による更新と式 (3.25) によるサンプルの生成を繰り返す．これにより，本手法で用いる B 個の基底形状ベクトルを一定の計算コストで増分的に算出することができる．これら全ての基底ベクトルを横に並べた行列を式 (3.1) における基底形状行列 \mathcal{M}_t とする．また，ここで同時に得られる固有値 $\lambda_{(1)} \sim \lambda_{(B)}$ の平方根を大きい順に並べた B 次元ベクトル μ も定義する． μ の各要素は，顔変形パラメータである a_t の各要素の標準偏差に対応する．それゆえ本手法では，後に述べるシステム雑音の分散や仮説の重みの決定に μ を使用している．

正確な頭部変形モデルを構築するためには、できるだけ信頼性の高い形状データを使用する必要がある。そこで本手法では、推定姿勢 p_t から判断して、ユーザがカメラの方を向いている場合の M'_t だけを分析対象として使用する。これは、ユーザが横を向いている場合と比較して特徴点追跡に成功している可能性が高いためである。また、サンプル形状 M'_t と前フレームまでの平均形状 \bar{M}_{t-1} との間の距離を計算し、その距離が一定以上であるという条件を満たす場合にのみ M'_t を CCIPCA への入力とする。実際に追跡の中で得られる顔形状データはほとんどが無表情の顔であるため、すべてのサンプルを均等に扱った場合、変形を含むデータの学習に対する影響が小さくなってしまう。それに対してこのような基準を加えることで、無表情のサンプルを学習の対象から除外している。

3.3 頭部変形モデルを用いた頭部姿勢推定

本節では、頭部変形モデルと左右2台のカメラからの入力画像を用いて、3次元な頭部姿勢を実時間で推定するための手法を説明する。これは図3.1の *Pose estimation step* での処理に対応する。

3.3.1 パーティクルフィルタを用いた頭部姿勢と顔形状の推定

本手法における頭部姿勢推定手法の基本的な枠組みは先行研究[43]に準ずる。ここでは3.2.1節で述べた頭部変形モデルとパーティクルフィルタを利用し、各入力画像フレーム t に対して $(6+B)$ 次元状態ベクトル $x_t = (p_t^T, a_t^T)^T$ を推定する。このとき、 p_t は世界座標系からモデル座標系への並進と回転を表す6次元頭部姿勢ベクトルである。

推定の流れを図3.6に示す。パーティクルフィルタ[18]では、状態ベクトルの確率密度関数を図3.6のように離散的な重み付きの仮説群により表現する。本手法で用いる仮説群 $\{(s_t^{(i)}; \pi_t^{(i)})\} (i=1 \dots N)$ は、 $(6+B)$ 次元状態空間内の N 個の仮説 $s_t^{(i)}$ と、各仮説に対応する重み $\pi_t^{(i)}$ で構成される。

まず、直前のフレーム $t-1$ の仮説群 $\{(s_{t-1}^{(i)}; \pi_{t-1}^{(i)})\}$ と次式の動作モデルを用いて新たな仮説を N 個生成する。

$$s_t^{(i)} = s'_{t-1} + \tau v_{t-1} + \omega \quad (3.26)$$

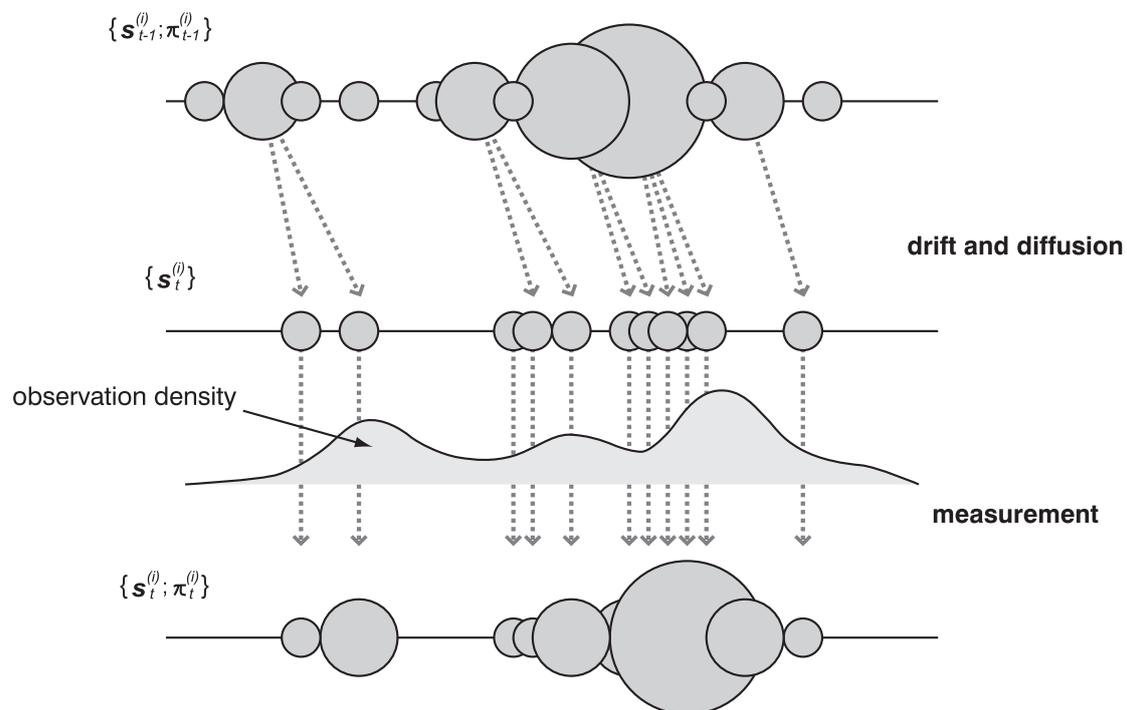


図 3.6: パーティクルフィルタにおける推定の流れ

この動作モデルでは隣り合うフレーム間での頭部姿勢の変化が等速直線運動に従うことを仮定している。 $s_{t-1}^{(i)}$ は $\{(s_{t-1}^{(i)}; \pi_{t-1}^{(i)})\}$ の中から選択された仮説であり、 τ はフレーム間の時間間隔、 v_{t-1} は直前のフレーム $t-1$ で計算されている状態ベクトル x の速度、 ω はシステム雑音である。

このとき、 ω の分布によって仮説 $s_t^{(i)}$ の拡散の性質が決定される。本手法では、 $(6+B)$ 次元ベクトル ω の各要素をそれぞれ固有の分散を持つ平均が 0 のガウス雑音としている。そして、これら $(6+B)$ 個の各要素の分散を並べたベクトル $\varsigma = (\varsigma_p^T, \varsigma_a^T)^T$ に関して、頭部姿勢パラメータに対応する部分である ς_p と顔変形パラメータに対応する部分である ς_a のそれぞれについて異なる手法で値を決定している。 ς_p に関しては、状態ベクトルの速度に応じて適応的に制御する。このような制御により、姿勢推定の精度を高く維持するとともに、ユーザの突発的な動作にも高い追従性で追跡することが可能となる [43]。一方 ς_a に関しては、顔変形パラメータ a_t の標準偏差に対応するベクトル μ を用いて $\varsigma_a = \kappa\mu$ とする。 μ の計算方法については 3.2.4 節で述べた通りである。なお、 κ は経験的に 0.5 に設定している。

次に、新たな仮説 $s_t^{(i)}$ に対応する重み $\pi_t^{(i)}$ を決定する．これは仮説と入力画像との一致度を示す量であり、各カメラからの入力画像に対する仮説の一致度 $\mathcal{N}_h(s_t^{(i)})$ をもとに、式 (3.27) のような関数により計算される．

$$\pi_t^{(i)} \propto \exp\left(-\frac{(2K - c_t^{(i)})^2}{2\sigma^2}\right) \exp\left(-\frac{1}{2} \sum_{b=1}^B \left(\frac{a_{t,b}^{(i)}}{\mu_b}\right)^2\right) \quad (3.27)$$

$$c_t^{(i)} = \sum_{h \in \{L, R\}} \mathcal{N}_h(s_t^{(i)}) \quad (3.28)$$

ここで、 $c_t^{(i)}$ は式 (3.28) に示すように全てのカメラにおける評価値 $\mathcal{N}_h(s_t^{(i)})$ の和であり、 $-2K$ と $2K$ の間の値を取る．式 (3.27) の第1項はこの $c_t^{(i)}$ をガウス関数により評価したもので、標準偏差 σ は経験的に3.0に設定している．一方、第2項は顔変形パラメータ $a_t^{(i)}$ についての関数であり、この項を乗じることでモデルが過度に変形しないように拘束を与えている．このとき、 $a_{t,b}^{(i)}$ は $a_t^{(i)}$ の b 番目の要素、 μ_b は μ の b 番目の要素である．以上の計算を行った上で、合計が1になるよう $\pi_t^{(i)}$ を正規化する．

なお、ここで用いられる評価関数 $\mathcal{N}_h(s_t^{(i)})$ は次のように定義される．まず、仮説 $s_t^{(i)}$ が与えられたとき、 $s_t^{(i)}$ の形状成分にあたる $a_t^{(i)}$ と式 (3.1) を用いて頭部モデルの形状を変形し、次いで、 $s_t^{(i)}$ の姿勢成分である $p_t^{(i)}$ によって変形後のモデルを移動する．その後、式 (3.4) と同じ投影関数を用いて K 個の3次元特徴点をカメラ h からの入力画像フレーム I_t^h に投影し、その投影点周辺の画像とテンプレート集合 T_h 中の対応するテンプレートとの間のマッチングスコアを正規化相関により計算する．このマッチングスコアを K 個の投影点について計算し、それらの和を $\mathcal{N}_h(s_t^{(i)})$ の出力値とする．

最後に、仮説群 $\{(s_t^{(i)}; \pi_t^{(i)})\}$ を用いて、現在の姿勢 p_t と顔変形 a_t を表す状態ベクトル x_t を計算する．ここでは、重みが最大となる仮説の近傍に属する仮説集合の加重平均により x_t を求める．

$$w_t^{(i)} = \begin{cases} 1 & \text{if } \|s_t^{(i)} - s_t^{(M)}\| < d \\ 0 & \text{else} \end{cases} \quad (3.29)$$

$$x_t = \frac{\sum_{i=1}^N s_t^{(i)} \pi_t^{(i)} w_t^{(i)}}{\sum_{i=1}^N \pi_t^{(i)} w_t^{(i)}} \quad (3.30)$$

このとき、 $s_t^{(M)}$ は最大の重みを持つ仮説である．現在のところ、 d の値は経験的に決定している．

また，次の画像フレームでの姿勢推定のために x_t の速度 v_t を計算しておく．

$$v_t = \frac{x_t - x_{t-1}}{\tau} \quad (3.31)$$

v_t の要素のうち，顔変形に対応する最後の B 個の要素は 0 に設定する．これは，顔変形のパラメータの変化は等速直線運動の仮定には十分に適合しないためである．

3.3.2 Halfway Partitioned Sampling

3.3.1 節で述べた手法により新たな仮説群 $\{(s_t^{(i)}; \pi_t^{(i)})\}$ を得ることは可能である．本手法ではさらに，高次元状態空間中の確率密度関数を限られた数の仮説でより効率良く近似するために，partitioned sampling [25] と似た考え方をを用いて仮説の生成・重み付けを行い， $\{(s_t^{(i)}; \pi_t^{(i)})\}$ を決定する．ここでは，この仮説の更新手法を halfway partitioned sampling と呼ぶこととし，その概要を以下に説明する．

ユーザの動きを注意深く観察すると，人間の頭部姿勢や顔形状の変化は大体 2 種類の型に分類できることがわかる．1 つは顔の変形をあまり伴わない頭部姿勢の剛体的な変化であり，もう 1 つは頭部姿勢の変化をあまり伴わずに顔形状を中心に变化するものである．そこで，両方の型に効率良く対処するために，まず，全仮説の半分には式 (3.26) のうち姿勢成分の変化だけを適用し，残りの半分には式 (3.26) の顔変形成分だけを適用する．その後，式 (3.28) と式 (3.27) によって仮説の重みを決定し，新たな仮説群 $\{(s_t^{(i)}; \pi_t^{(i)})\}$ を得る．さらに， $\{(s_t^{(i)}; \pi_t^{(i)})\}$ に対して通常のリサンプリング処理，すなわち，頭部姿勢と顔変形を合わせた全体的な状態空間の中での仮説の再生成・重み付けを行う．以上の処理により得られた $\{(s_t^{(i)}; \pi_t^{(i)})\}$ は，状態ベクトル x_t の確率密度関数を比較的少量の仮説で効率良く表現できる．また，頭部姿勢運動と顔変形が同時に発生した場合についても，上記のリサンプリング処理の適用により適切に対処することが可能となっている．

3.4 評価実験

提案手法の性能を評価するために実験を行った．本実験では，Intel Pentium4 3.0GHz と Windows XP を搭載した汎用 PC を 1 台使用した．入力画像として，2 台の IEEE1394 カメラ (Point Grey Research 社製 Flea (図 3.7)) により取り込ま

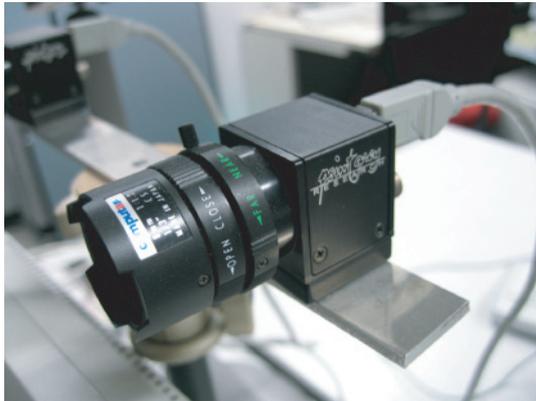


図 3.7: 実験で利用したカメラ



図 3.8: 頭部姿勢推定システムの概観

れた 640×480 画素の画像を使用した。カメラはあらかじめ、Zhang らの手法 [39] を利用してキャリブレーションを行っている。また、画像テンプレートの大きさは 16×16 画素、パーティクルフィルタの仮説の総数は 1000 であった。このとき、本システムは毎秒 30 フレームで動作した。システム全体の概観は図 3.8 に示すとおりである。

本実験で用意した画像列では、ユーザが頭部を動かしながら顔の変形を時折行っている。この画像列は 60 秒 (1800 フレーム) 分のデータで構成されている。このうち最初の 120 フレームでは、初期化時に自動的に獲得される頭部剛体モデルを用いて頭部姿勢を推定した。これは、姿勢推定の開始直後は、CCIPCA によって計算される基底行列 M の信頼性が低いためである。その後の残り 1680 フレームでは、逐次的に更新される頭部変形モデルにより頭部姿勢と顔変形を推定した。なお、この 1680 フレームのうちの最初の約 1100 フレームでは、ユーザは口の開閉やしかめ面といった顔変形を行いながら頭部運動を行っている。これに対し、残りの約 600 フレームでは、ユーザは顔変形をほとんど行わずに頭部の剛体運動だけを行っている。

また、上記の推定結果と比較する目的で、同じ 1800 フレームに対して、頭部剛体モデルを用いた頭部姿勢推定も行った。この頭部剛体モデルは、2.2 節で述べた初期化部の処理により構築されたものであり、変形モデルを用いる実験の最初の 120 フレームで使用するものと同じモデルである。このようにして得られた 2 種類の頭部姿勢推定結果を比較した。

図 3.9 に頭部姿勢推定の結果画像を示す。この図には、推定された頭部姿勢に対

応するモデル座標軸と，推定された顔形状 M_i を画像平面に投影した点が描かれている．図の左列が頭部剛体モデルを用いた場合の結果画像，右列が頭部変形モデルを用いた場合の結果画像である．これらの結果画像から，提案手法により得られた頭部変形モデルが顔変形への対処に大きく貢献していることがわかる．

さらに，図 3.10 には頭部剛体モデルと頭部変形モデルをそれぞれ用いた場合の推定結果のグラフを示す．この図において，細い線は剛体モデルを用いた場合の推定結果であり，太い線は変形モデルを用いた場合の推定結果である． x は水平方向， y は鉛直方向， z は深さ方向の運動であり， $roll$ は z 軸回り， yaw は y 軸回り， $pitch$ は x 軸回りの回転である．この図を見ると，第 121 フレームから第 1200 フレーム付近までは両方の推定結果に明確な相違が見られる．図 3.9 の推定結果画像と図 3.10 中の対応するフレームでの推定結果を考慮すると，頭部変形モデルを用いた方がより正しく推定できていることがわかる．すなわち，頭部変形モデルは，剛体モデルとは異なり，顔形状の変化に対して適切に対処可能であることが示された．一方，残りの約 600 フレームでは両方の推定結果がほぼ一致している．このことから，頭部変形モデルが不要な顔変形を伴うことなく，正しく頭部剛体運動を推定できていることがわかる．

また，図 3.10 中最下部のグラフは，頭部変形モデルで用いる基底形状ベクトルに対応する 5 個の固有値の累積寄与率を示す．累積寄与率は，低次元のモデルがデータ全体の情報をどの程度まで表現できるかを示す一つの目安である．本手法のように CCIPCA を用いて基底を更新する場合，新たな変形情報が得られるとデータのばらつきが大きくなるため一時的に累積寄与率は低下するが，学習による基底の更新を繰り返すことで累積寄与率は上昇し，再びデータ全体を表現することが可能になる．実験では追跡開始直後に一度大きく累積寄与率が低下しているが，その後の学習により 600 フレーム（20 秒）程度で 80% 近くまで回復していることがわかる．

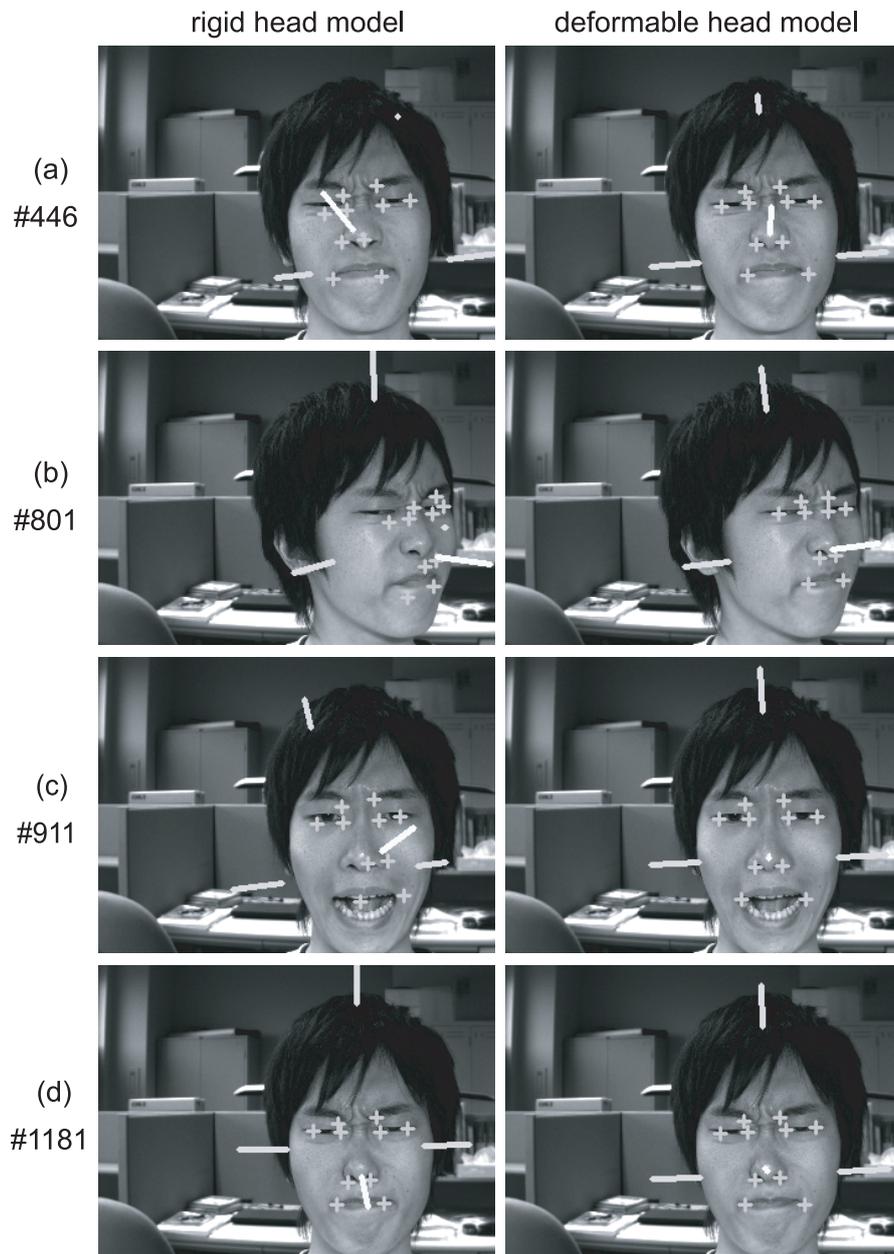


図 3.9: 頭部姿勢推定の結果画像

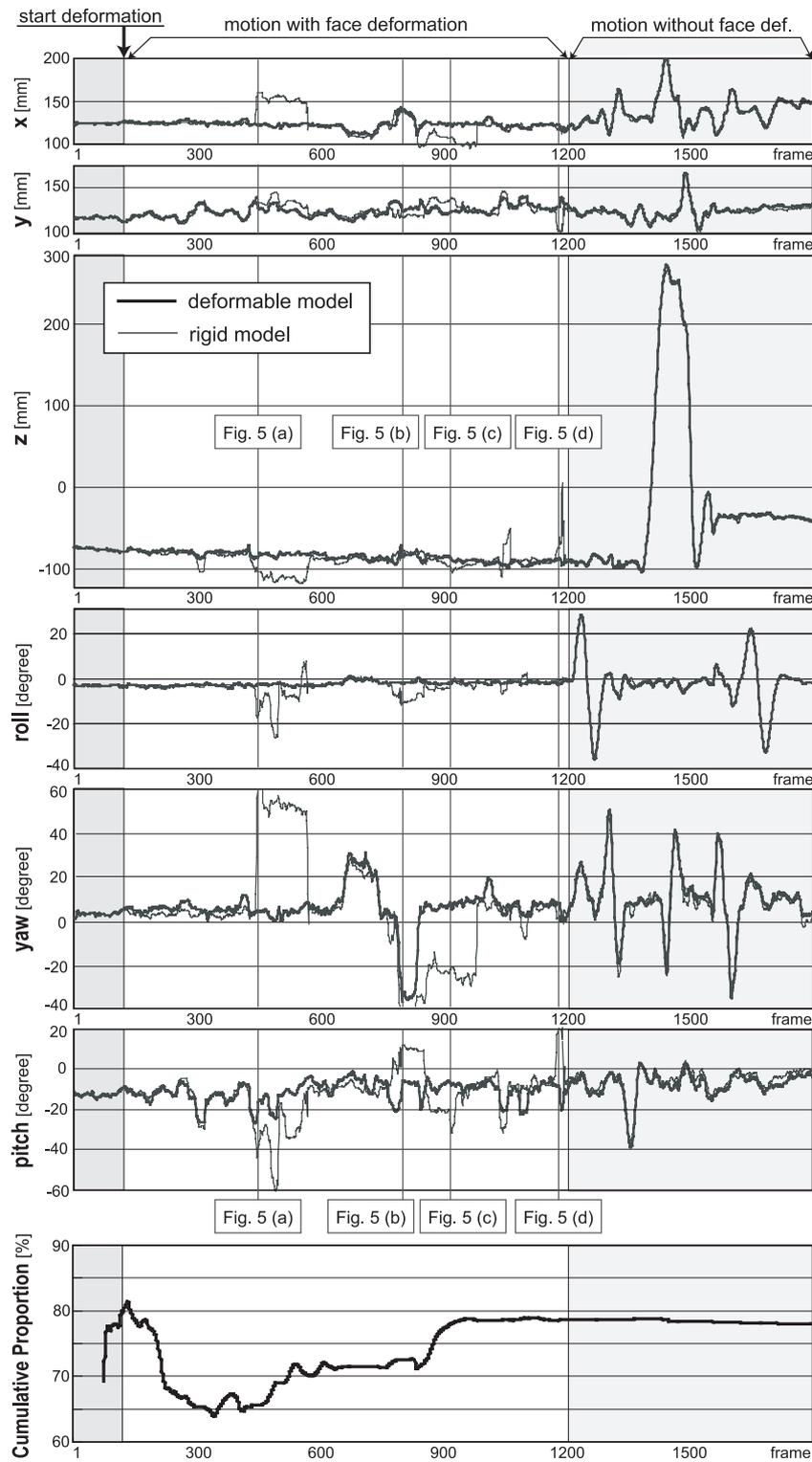


図 3.10: 頭部剛体/変形モデルを用いた頭部姿勢推定の結果

第4章 パラメータ分離モデルを用いた単眼による頭部姿勢推定

4.1 はじめに

第3章で述べた手法では、システムの構築に2台以上のカメラを必要としていた。複数台のカメラを用いるシステムでは、カメラ画像から直接追跡対象の3次元位置及び形状を算出できるほか、複数の視点から対象を捉えることで追跡の安定性も向上するという利点がある。しかし、実際の応用を考えた場合、スペースの確保や費用の面で複数台のカメラを設置するのは難しい場合が多い。近年では、ビデオチャットやテレビ電話などの普及とカメラの低価格化に伴い、大規模な設置型システムに留まらず、個人用のデスクトップ環境やモバイルデバイスなど、インタラクション技術をより多様な場面に適用することが可能になってきている。こうした中で幅広いアプリケーション応用を想定した頭部姿勢推定システムを構築する上では、1台のカメラのみを用いてシステムを構築できる技術が要求される。本章では、追跡の精度・安定性を損なうことなく、さらに設置コスト面での欠点も補うような、1台のカメラで構築可能な3次元頭部姿勢推定システムを提案する。

単眼で3次元姿勢を推定するために、本手法では事前学習により構築した人間の顔形状に関するモデルを利用して状態推定を行う。第2章で述べたとおり、顔形状の変動における要因は大きく次の二つに分けて捉えることができる。

- 眉の上下や口の開閉といった、個人内の変形に対応する要素
- 目の位置や鼻の高さといった、個人差に対応する要素

これらを全て単一のパラメータで表現することで特定のユーザに依存しない汎用性の高いモデルを作成する場合、パラメータの次元数を低く抑えることは困難となる。特に実時間の推定を行う場合、安定性や計算速度の面でのデメリットが大きい。これに対し、二つの要素をそれぞれ異なるパラメータにより記述したモデ

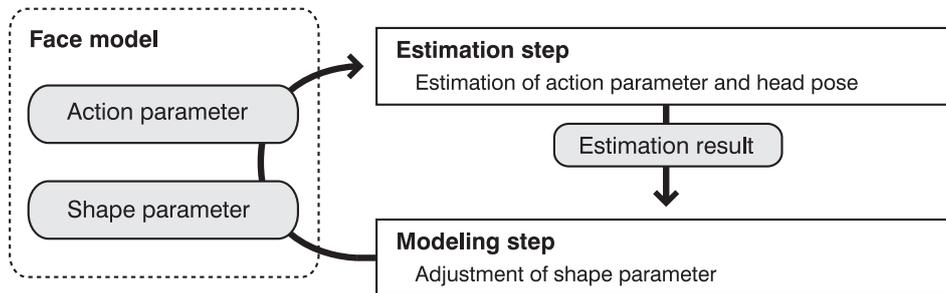


図 4.1: 単眼システムの概要

ルは、近似精度を保ったまま個々のパラメータの次元を低く抑えることが可能になる。さらに形状変化の要因を分けて捉えることができるため、表情認識などの応用を考える上でも有用であると言える。

本章では、先に述べた3つの条件を満たす頭部姿勢推定を実現するための枠組みとして、顔形状の個人内変動（変形パラメータ）と個人間変動（個人差パラメータ）を分離して表現するパラメータ分離モデルを用いた頭部姿勢推定手法を提案する。本手法では、分離モデルの元で二つの異なる処理を統合することで実時間頭部姿勢推定を実現する。一つはパーティクルフィルタを利用した姿勢、変形の時系列推定であり、フレームごとに変化する頭部姿勢と変形パラメータの安定した追跡を可能とする。もう一つはバンドル調整の枠組みを利用した個人差の調整であり、複数フレームの情報をを用いたパラメータの最適化を実時間処理の中で逐次的に実行する。このような二つの手法を統合することで、二つのパラメータの性質の違いに対してそれぞれ適切なアプローチによる推定を実現することができる。これにより、任意のユーザに対して事前の準備を伴うことなく、顔変形を含む頭部姿勢推定が単眼でも可能になる。そして、複数のカメラを用いた手法と比較しても遜色の無い頭部姿勢推定が実現できることを、評価実験により検証する。

システムの概要を図4.1に示す。すなわち、変形と姿勢の時系列推定を行う *Estimation step* と、複数フレームに対する最適化により個人差の調整を行う *Modeling step* により構成される。以下、本章の流れは次のようになる。第4.2節では、本手法で用いるパラメータ分離モデルの作成方法について述べる。

4.2 頭部変形モデルの構築

本節では、本手法で用いる顔形状モデルの詳細とその作成方法について述べる。サンプルとして用意した形状データをもとに、任意のユーザに対応できる汎用的な顔形状モデルを作成する。モデル作成を行う上で要因ごとにパラメータを分離する手法はいくつか考えられるが、本研究では、Vlasic らの手法と同様に N-mode SVD にもとづく多重線形モデルを利用した [34]。こうした多重線形モデルは Vasilescu らが顔認識やテクスチャ生成に応用したことでコンピュータビジョンの分野でも近年注目を集めた手法であり [32, 33]、複数人の変形のような多数の要因によって形成される対象を、より洗練された形でモデル化することができる。

4.2.1 モデル構築に用いるテンソル解析

まず、具体的なモデル構築について述べる前に、本手法で用いるテンソル解析の基本について解説する [22]。

テンソル

テンソルとはベクトルや行列といった多次元配列を一般化した構造であり、スカラー、ベクトル、行列は、それぞれ 0 階、1 階、2 階のテンソルに対応する。図 4.2 に示す \mathcal{A} は、 $I_1 \times I_2 \times I_3$ 個の要素を持つ 3 階のテンソルを表している。

テンソルの展開と行列との積

行列における行と列に対応するような要素の方向を、テンソルでは「モード」と呼ぶ。 N 階のテンソル $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ において、モード n の要素ベクトルとは n 番目のインデックスのみを変更し、それ以外を固定した時に得られる I_n 次元のベクトルを指す。このとき、 \mathcal{X} におけるモード n のベクトルすべてを列方向に並べた行列 $X_{(n)} \in \mathbb{R}^{I_n \times (I_1 I_2 \dots I_{n-1} I_{n+1} \dots I_N)}$ を定義することができる。図 4.2 の右側は、3 階のテンソル \mathcal{A} を各モードで展開した時の様子を示す。

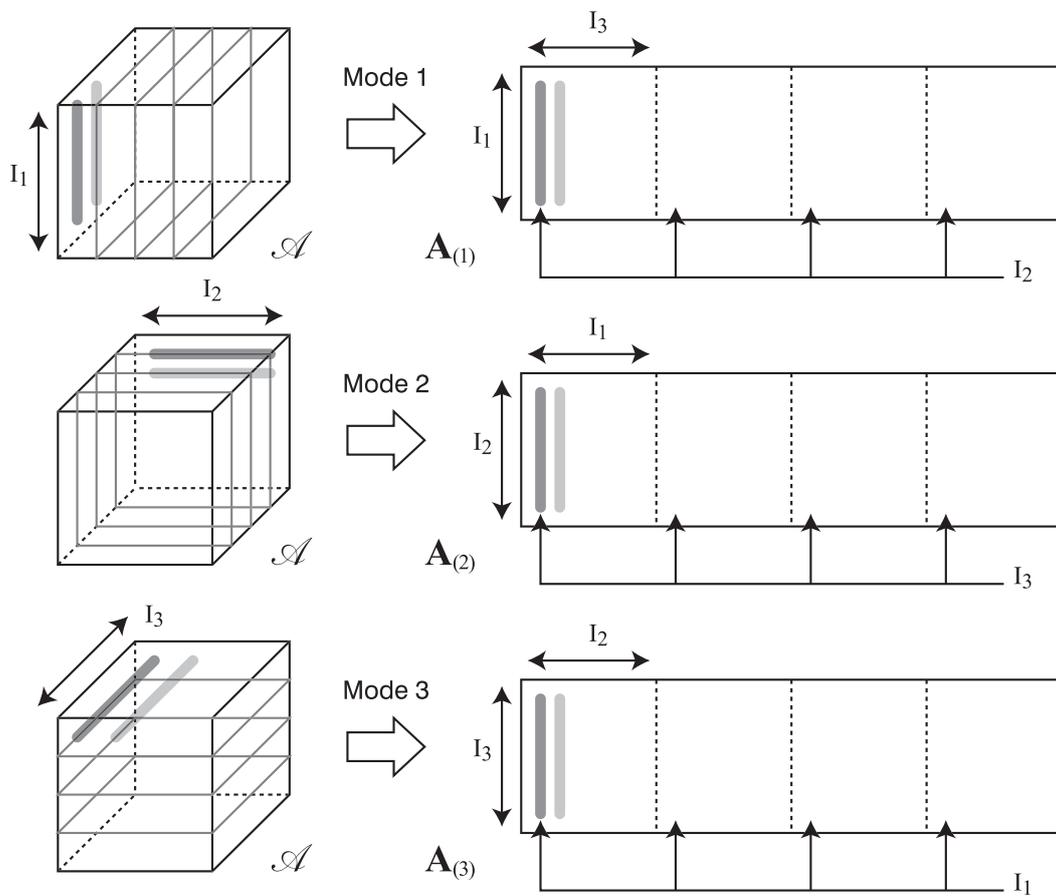


図 4.2: 3 階のテンソルの展開

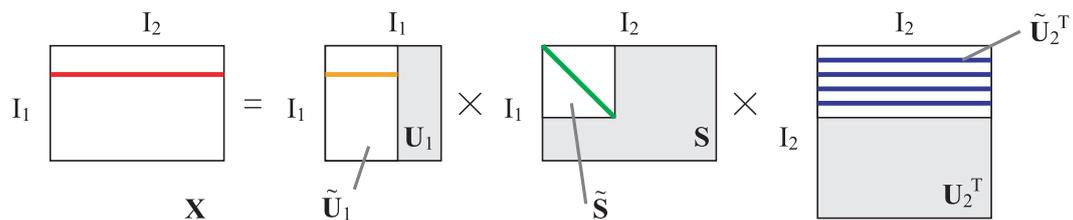


図 4.3: Singular Value Decomposition

さらに，テンソル $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ と行列 $C \in \mathbb{R}^{J_n \times I_n}$ のモード n における積 (Mode- n product) $\mathcal{Y} = \mathcal{X} \times_n C$ は次のように定義される．

$$\mathcal{Y}_{i_1 \dots i_{n-1} j_n i_{n+1} \dots i_N} = \sum_{i_n} \mathcal{X}_{i_1 \dots i_{n-1} i_n i_{n+1} \dots i_N} C_{j_n i_n} \quad (4.1)$$

つまり， \mathcal{Y} のモード n 空間は \mathcal{X} のモード n 空間の行列 C による線形写像となっており，これはテンソルの展開を用いて次のように書くことができる．

$$\mathbf{Y}_{(n)} = \mathbf{C} \mathbf{X}_{(n)} \quad (4.2)$$

N-mode SVD

まず，通常の SVD (Singular Value Decomposition) について説明する． $I_1 \times I_2$ 行列 X は，必ず次式のような積の形に分解することができ，これを Singular value decomposition (特異値分解) と呼ぶ．

$$\mathbf{X} = \mathbf{U}_1 \mathbf{S} \mathbf{U}_2^T \quad (4.3)$$

各 \mathbf{U}_n は $I_n \times I_n$ 直交行列であり， \mathbf{S} は対角要素以外は 0 である次のような $I_1 \times I_2$ 行列である．

$$\mathbf{S} = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_{\min(I_1, I_2)}) \quad (4.4)$$

σ_i は特異値と呼ばれ， σ_1 から大きい順に並べられる．また，このとき $\mathbf{U}_1, \mathbf{U}_2$ の列ベクトルは順に左特異値ベクトル，右特異値ベクトルと呼ばれ，それぞれ行列 X の列空間 (モード 1)，行空間 (モード 2) の正規直交基底になっている．したがって，各々の基底の次元数を削減することによって，次のように行列 X の低次元近似表現を得ることができる．

$$\mathbf{X} \approx \check{\mathbf{U}}_1 \check{\mathbf{S}} \check{\mathbf{U}}_2^T \quad (4.5)$$

以上の概略を図4.3に示す．図中に色付きの線で示したように，例えば U_1 のある1行（橙）を取り出して特異値（緑）で重み付けされた基底（青）に掛け合わせることで， X の対応した行ベクトル（赤）が得られる．つまり， U_1, U_2 の行ベクトルは各モードの特定のデータに対応したパラメータ（基底ベクトルに対する重み）と考えることができる．

このような行列におけるSVDを，さらに3階以上のテンソルに対して一般化したものを，N-mode SVD または HOSVD (Higher Order SVD) と呼ぶ．

式4.3は次のようにテンソルの積の形で書くことができる．

$$\mathbf{X} = \mathbf{S} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \quad (4.6)$$

N 階のテンソル \mathcal{X} に対してもこれと同じように，次のような分解を行うことができる．

$$\mathcal{X} = \mathcal{C} \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \dots \times_N \mathbf{U}_N \quad (4.7)$$

テンソル \mathcal{C} はコアテンソルと呼ばれ，次式により算出される．

$$\mathcal{C} = \mathcal{X} \times_1 \mathbf{U}_1^T \times_2 \mathbf{U}_2^T \dots \times_N \mathbf{U}_N^T \quad (4.8)$$

式(4.3)における \mathbf{S} とは異なり，コアテンソルは対角成分以外にも値を持つ．

U_n は次のように，テンソル \mathcal{X} をモード n で展開した行列 $\mathbf{X}_{(n)}$ に対するSVDにより得られる．

$$\mathbf{X}_{(n)} = \mathbf{U}_n \mathbf{S}_n \mathbf{V}_n^T \quad (4.9)$$

すなわち，行列の場合と同様に U_n の列ベクトルはモード n 空間の正規直交基底になることがわかる．したがって，次のようなテンソル \mathcal{X} の低次元近似表現が得られる．

$$\mathcal{X} \approx \mathcal{C} \times_1 \check{\mathbf{U}}_1 \times_2 \check{\mathbf{U}}_2 \dots \times_N \check{\mathbf{U}}_N \quad (4.10)$$

ただし，N-mode SVD においては，単純な下位基底の切り捨てだけでは最良の近似が得られないことが知られており，ALS (Alternating Least Squares) と呼ばれるアルゴリズムを用いて $\check{\mathbf{U}}_n$ の計算を行うことでより良い近似が可能になる [23] ．

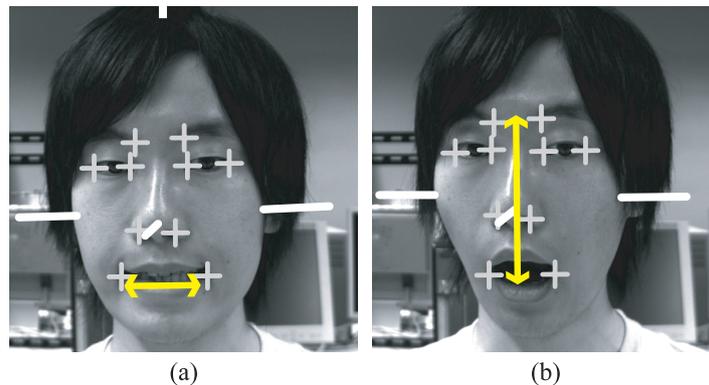


図 4.4: モデル化の対象となる顔変形の例

4.2.2 N-mode SVD による形状パラメータの分離

次に、具体的なモデル構築作業について述べる。

学習に用いるデータを取得する際には、第3章で述べた手法にもとづく頭部姿勢推定を利用した。顔形状モデルも第3章で述べた手法と同じものを採用しており、 K 個の特徴点の3次元座標で構成された $3K$ 次元形状ベクトル M とそれに対応付けられた各特徴点のテンプレート画像により表現される。本手法では、この形状ベクトル M を N-mode SVD による解析を利用して事前にモデル化する。

4.5 章で述べる本論文の実験では、モデル化の対象とする変形は次の二種類に絞った。一つは図 4.4(a) のような口の横方向の運動であり、もう一つは同図 (b) のような眉の上下を含む顔全体の縦方向の運動である。モデル構築のサンプルとして用いる S 人の人物それぞれについてこれらの運動を行う様子を一定時間記録し、以下に述べる手順で自動的に形状データを抽出する。

それぞれの運動には基準となる距離 l (図 4.4 中の矢印に対応) を定め、まず人物毎に l の最大値 l_{max} と最小値 l_{min} を計測する。これにより定義される変形率 $r = (l - l_{min}) / (l_{max} - l_{min})$ を基準にして、 $r = 0.0 \sim 1.0$ の範囲で等間隔に、合計 A 通りの形状データを取得する。

さらに、取得した形状データは顔中心で左右に分割し、それぞれを反転させて生成した形状を二つの独立したサンプルとして扱う。これによりモデルを左右対称なものとし、不要な回転成分が含まれるのを防いでいる。また、本手法では $3K \times 2S \times A$ 個のサンプル全体の平均 \bar{M} を差し引き、正規化したデータに対する解析を行う。

このようにして得られたデータを元に、図 4.5 のようなデータテンソル $\mathcal{T} \in$

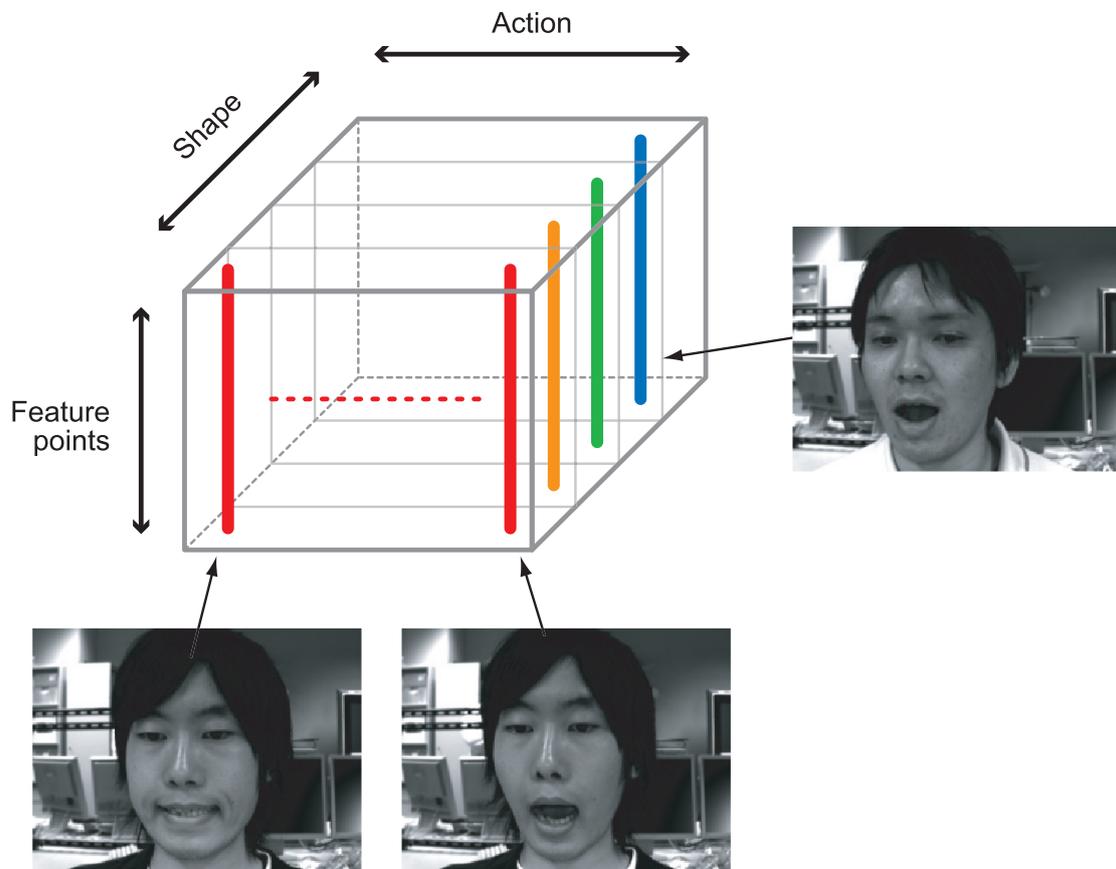


図 4.5: モデル構築に用いるデータテンソル

$\mathbb{R}^{3K \times 2S \times A}$ を作成する．図中，Feature points 方向には顔形状ベクトル M が対応し，Action 方向には同一人物のデータが，Shape 方向には r の値が等しい変形が，それぞれ対応付けられた形で格納される．

前節で説明した N-mode SVD を用いてテンソルを分解することで，データテンソル \mathcal{T} は次のように記述できる．

$$\mathcal{T} = \mathcal{C} \times_{\text{feature}} \mathbf{U}_{\text{feature}} \times_{\text{shape}} \mathbf{U}_{\text{shape}} \times_{\text{action}} \mathbf{U}_{\text{action}} \quad (4.11)$$

ここで $\mathcal{M} = \mathcal{C} \times_{\text{feature}} \mathbf{U}_{\text{feature}}$ とおくと，次のような表現が得られる．

$$\mathcal{T} = \mathcal{M} \times_{\text{shape}} \mathbf{U}_{\text{shape}} \times_{\text{action}} \mathbf{U}_{\text{action}} \quad (4.12)$$

\mathcal{M} は特徴点モード空間，すなわち形状ベクトルに関する基底を含むテンソルであり，実際には次のように直接 \mathcal{M} を計算することができる．

$$\mathcal{M} = \mathcal{T} \times_{\text{shape}} \mathbf{U}_{\text{shape}}^T \times_{\text{action}} \mathbf{U}_{\text{action}}^T \quad (4.13)$$

さらに，ALS アルゴリズムを用いて次元数を削減した基底 ($\check{\mathcal{M}} \in \mathbb{R}^{3K \times S' \times A'}$ ， $\check{\mathbf{U}}_{\text{shape}} \in \mathbb{R}^{2S \times S'}$ ， $\check{\mathbf{U}}_{\text{action}} \in \mathbb{R}^{A \times A'}$) を用いることで，次のようにデータテンソルの近似表現が得られる．

$$\mathcal{T} \approx \check{\mathcal{M}} \times_{\text{shape}} \check{\mathbf{U}}_{\text{shape}} \times_{\text{action}} \check{\mathbf{U}}_{\text{action}} \quad (4.14)$$

通常の SVD との対応から明らかなように， $\check{\mathbf{U}}$ の行ベクトルはデータテンソルに含まれる特定の変形や個人に対応する係数となっている．そこで本手法では， $\check{\mathbf{U}}_{\text{shape}}$ を元に個人差を表す係数の平均 \bar{s} 及び標準偏差 σ_s を，また $\check{\mathbf{U}}_{\text{action}}$ を元に変形を表す係数の平均 \bar{a} 及び標準偏差 σ_a を，それぞれ算出する．この分布を元に個人差パラメータ $s \in \mathbb{R}^{S'}$ と変形パラメータ $a \in \mathbb{R}^{A'}$ を与えることで，次のように任意の顔形状ベクトル M を記述することができる．

$$M = \bar{M} + \check{\mathcal{M}} \times_{\text{shape}} s^T \times_{\text{action}} a^T \quad (4.15)$$

本手法ではパラメータが正規分布に従うことを仮定し，第 4.4 節において平均と標準偏差をシステム雑音の大きさやパラメータ拘束範囲の決定にこれらの平均と標準偏差を利用している．

生成される顔形状の例を図 4.6 に示す．図 4.6 において縦方向では個人差パラメータが，横方向では姿勢及び変形パラメータが一致している．

すなわち，以上の処理により，顔形状のモデルとして以下の情報が得られる．

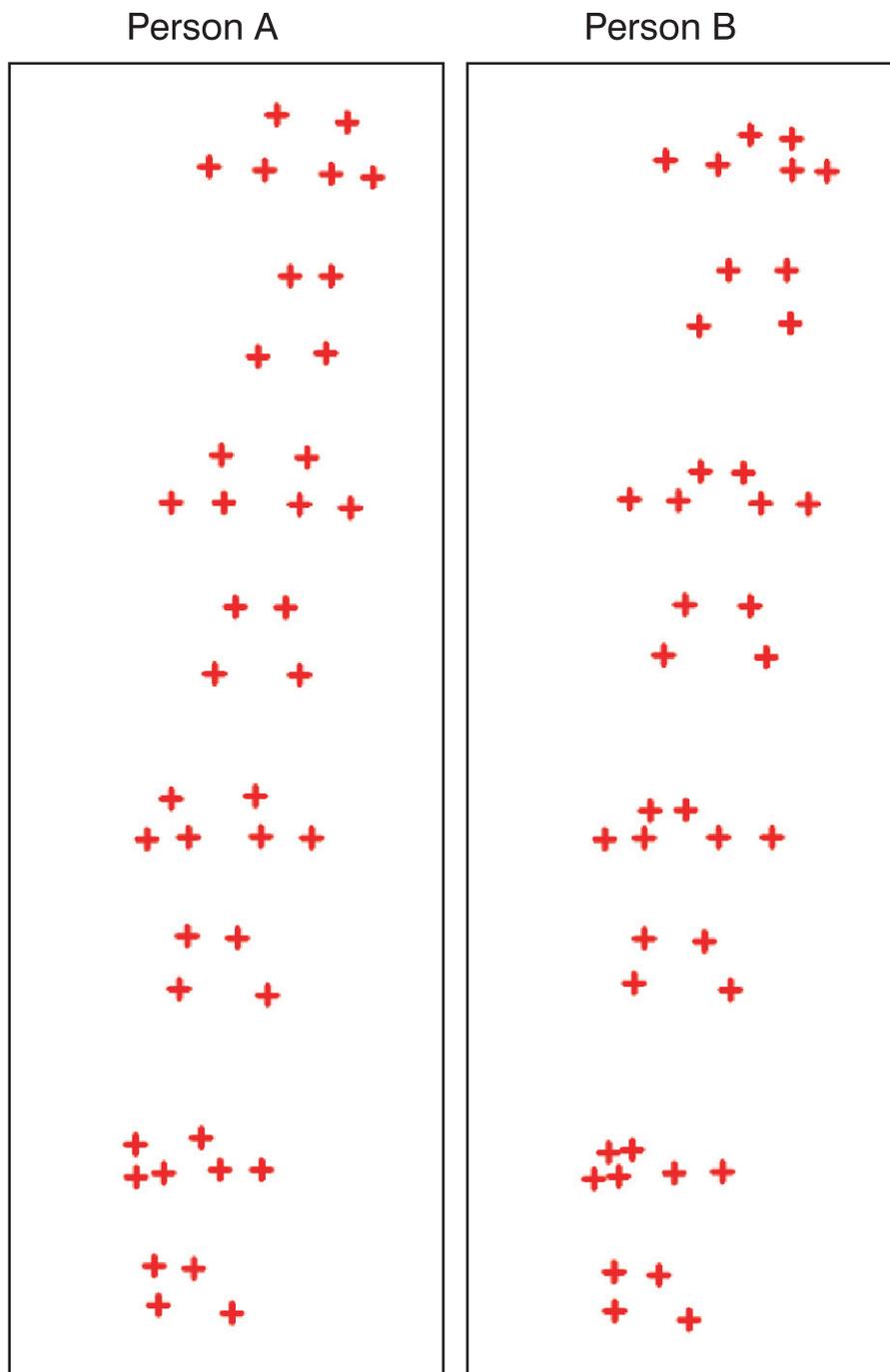


図 4.6: パラメータ分離モデルにより生成される顔形状の例

- モデルテンソル $\tilde{\mathcal{M}}$
- 個人差パラメータ分布 \bar{s}, σ_s
- 変形パラメータ分布 \bar{a}, σ_a

このモデルを用いることで、ある時刻 t におけるユーザの状態は変形パラメータベクトル a_t と個人差パラメータベクトル s に世界座標系からモデル座標系への並進と回転を表す6次元頭部姿勢ベクトル p_t を加えた3つのパラメータで記述できる。このとき、各パラメータの添え字に表したように、変形 a_t と姿勢 p_t は時間に依存して常に変化する。一方、個人差 s は同じユーザに対して時間に関わらず一定の値をとる。本手法では、こうしたパラメータの性質の違いに合わせ、以下の節で述べる二つの手法を統合した頭部姿勢推定を行う。

4.3 バンドル調整の枠組みによる個人差パラメータ推定

本節では、4.2節で説明したモデルに対して個人差の調整を行う処理について述べる。これは図4.1における *Modeling step* の処理に相当する。

バンドル調整は最尤推定手法の一つであり、複数フレームに跨る誤差関数を最小化することでカメラ姿勢とモデル形状を同時に調整する。映像から3次元モデルを復元する Structure from Motion 技術などにおいて、おおまかな推定形状を精緻化するための手法として広く用いられている。近年、バンドル調整の枠組みにパラメータ表現された事前構築モデルを取り入れるモデルベースのバンドル調整手法が提案されている [11, 29]。Xin らはこれを利用して、安価なカメラからのビデオ入力を元に効率よく頭部の3次元形状を自動構築する手法を提案している [38]。Vachetti らの手法 [31] ではこの枠組みを実時間の頭部姿勢追跡に応用することで、任意のユーザに対して頭部の剛体運動推定を可能にしている。また、非剛体の Structure from Motion において、バンドル調整の枠組みを利用して基底を含めた最適化を行う例もいくつか提案されている [1, 9]。

本研究ではこのバンドル調整の枠組みをパラメータ分離モデルに導入することで、変形を含む形状推定を実時間処理の中で実現している。さらに、パラメータの制約条件を用いることで、逐次的なバンドル調整を行う上でより効果的な処理が可能となる。以下、その具体的な処理について説明する。

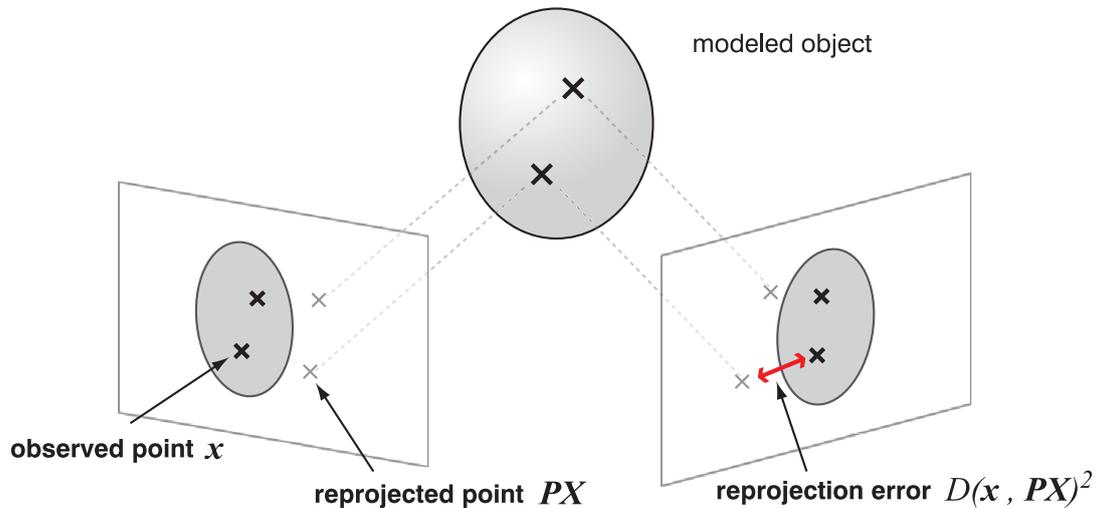


図 4.7: バンドル調整の概要

4.3.1 バンドル調整の問題設定

まず，バンドル調整は一般的に次のような誤差関数の最小化問題となる．

$$\min_{P_i, X_k} \sum_{i=1}^x \sum_{k=1}^n D(x_{ik}, P_i X_k)^2 \quad (4.16)$$

ここで X_k はモデル中 k 番目の点の3次元座標であり， P_i は i 番目のフレームにおける投影行列， x_{ik} は，フレーム i の入力から画像特徴により得られた点 k の正確な2次元座標を示す． $D(x_{ik}, P_i X_k)^2$ は，画像中の特徴点 x_{ik} とモデルにもとづく投影点 $P_i X_k$ とのユークリッド距離である．すなわち式 (4.16) は図 4.7 のように，動画フレームから得られた特徴点の2次元軌跡を元にして，投影点が軌跡に最も近くなるようなモデルの3次元形状とカメラ姿勢を推定する式になっている．

本手法で用いる顔形状モデルを用いた場合，バンドル調整によるパラメータ調整は次のようになる．まず，フレーム i におけるユーザ頭部の状態は，世界座標系からモデル座標系への並進と回転を表す6次元頭部姿勢ベクトル p_i と変形パラメータベクトル a_i ，そして時間に依存しない個人差パラメータベクトル s によって記述できる．このとき，次のような投影関数 \mathcal{P} を定義することができる．これは， a_i ， s から式 (4.15) によって決まる顔形状ベクトル M_i に頭部姿勢 p_i に応じた並進や回転を適用した上で， M_i の各特徴点を画像平面に投影する関数である．

$$m_i = \mathcal{P}(p_i, M_i) \quad (4.17)$$

すなわち, m_i は K 個の投影点の2次元座標で構成される $2K$ 次元ベクトルとなる.

一方, フレーム i における真の特徴点2次元座標を \hat{m}_i とする. これは *Estimation step* において姿勢推定の結果を元に探索した座標であり, 計算方法の詳細に関しては4.4節で述べる. すると, 第 t フレームにおいて次のような過去 n フレーム分の誤差関数が定義できる.

$$F_t = \sum_{i=t-n+1}^t D(\hat{m}_i, \mathcal{P}(p_i, M_i))^2 \quad (4.18)$$

本手法では Zhang らの手法 [41] と同様に, 新たなフレームが得られるごとに F_t の最小化を行う. これにより姿勢 p_t と変形 a_t の推定値を最適化すると共に, 個人差 s を更新していく.

Levenberg-Marquardt アルゴリズムによる誤差関数の最小化

バンドル調整における誤差関数の最小化には, 一般的に Levenberg-Marquardt アルゴリズム (LM アルゴリズム) が用いられる. これはガウスニュートン法に最急降下法を組み合わせたアルゴリズムであり, 収束の速さと結果の安定性を兼ね備えた, 非線形の最小化問題を解く上で標準的な手法になっている. まず, 簡単にアルゴリズムの概要について説明する. 本研究では, Lourakis らにより公開されている C 言語用の LM アルゴリズムライブラリ [24] を利用して以下の処理を実装した.

ここでは次式のような誤差関数の最小化問題を考える.

$$\begin{aligned} F &= \sum_i D(y_i, f(x_i))^2 \\ &= \|y - f(x)\|^2 \end{aligned} \quad (4.19)$$

ここで x はパラメータベクトル, y は観測ベクトルであり, 関数 f はパラメータと観測の結びつきを表す. つまり, 式 (4.18) では調整対象となる全てのパラメータを並べたベクトル $(p_{t-n+1}^T, \dots, p_t^T, a_{t-n+1}^T, \dots, a_t^T, s^T)^T$ が x に, 全フレームの特徴点観測座標を並べたベクトル $(\hat{m}_{t-n+1}^T, \dots, \hat{m}_t^T)^T$ が y に対応している.

LM アルゴリズムでは一般的な最小化アルゴリズムと同様に, パラメータの差分 δ_x を繰り返し計算することで最適化を行う. x の近傍でテイラー展開を行うことで, f に関して次のような近似が得られる.

$$f(x + \delta_x) \approx f(x) + J\delta_x \quad (4.20)$$

ここで J は x に関する f のヤコビアン $\frac{\partial f(x)}{\partial x}$ である．これを式 (4.19) に代入すると次式が得られる．

$$\begin{aligned} F &= \|\mathbf{y} - f(\mathbf{x}) + \mathbf{J}\delta_x\|^2 \\ &= \|\boldsymbol{\varepsilon} + \mathbf{J}\delta_x\|^2 \end{aligned} \quad (4.21)$$

ここで $\boldsymbol{\varepsilon} = \mathbf{y} - f(\mathbf{x})$ は誤差ベクトルを表す．したがって， $\frac{\partial F}{\partial \delta_x} = 0$ とおくことで，次式の解として δ_x を算出することができる．

$$\mathbf{J}^T \mathbf{J} \delta_x = -\mathbf{J} \boldsymbol{\varepsilon} \quad (4.22)$$

式 (4.22) は正規方程式と呼ばれ，ガウスニュートン法ではこれを利用した最小化を行う．ガウスニュートン法は解の近傍において高速な収束を実現できる一方，初期値が解から離れている場合は局所解に陥る可能性が高くなってしまう．LM アルゴリズムではこの欠点を解決するために，式 (4.22) を次のように変更した拡張正規方程式を用いる．

$$(\mathbf{J}^T \mathbf{J} + \mu \mathbf{E}) \delta_x = -\mathbf{J} \boldsymbol{\varepsilon} \quad (4.23)$$

ここで \mathbf{E} は単位行列であり， μ はダンピングファクターと呼ばれる非負の値である．式 (4.23) を用いることで， μ が大きい場合は次のように最急降下法に近い値が得られる．

$$\delta_x \simeq -\frac{1}{\mu} \mathbf{J} \boldsymbol{\varepsilon} \quad (4.24)$$

最急降下法では収束が遅くなるというデメリットがあるが，局所解に陥るのを避けることができる．一方， μ が 0 に近い場合は式 (4.22) を用いたガウスニュートン法に近い値が得られることになる．LM アルゴリズムでは，繰り返しの中で現在の状態に応じて μ の値を調整する．これにより，解から離れた初期状態では最急降下法を用いた探索を，解の近傍ではガウスニュートン法を用いた探索を行うことができる．

本手法ではさらに，LM アルゴリズムにおいてパラメータの調整範囲に対して制約条件を設定する [19]．

$$\min_x F, \quad \mathbf{x} \in \mathbf{C} \quad (4.25)$$

C はパラメータ x に対する制約範囲であり，本手法では次式のようなボックス型の拘束を考える．

$$C = \{x \mid x_{\min} \leq x \leq x_{\max}\} \quad (4.26)$$

x_{\min} , x_{\max} はパラメータの最小値，最大値を表す定数ベクトルとする．また，次のように x を C の範囲内に投影する関数 P_C を定義する．

$$P_C(x) = \begin{cases} x_{\max} & (x > x_{\max}) \\ x & (x_{\min} \leq x \leq x_{\max}) \\ x_{\min} & (x < x_{\min}) \end{cases} \quad (4.27)$$

最小化の際は，最初に式 (4.23) を用いて制約のない状態で差分 δ_x を算出した上で，式 (4.27) を用いて次のように定義される \hat{x} を解とする．

$$\hat{x} = P_C(x + \delta_x) \quad (4.28)$$

実際のアルゴリズムでは，式 (4.28) により求めたパラメータで誤差が大きくなってしまう場合，ラインサーチや最急降下法を組み合わせることで必ず誤差が小さくなるようにパラメータを更新していく．

4.3.2 バンドル調整の逐次実行

次に本節では，本手法における具体的なバンドル調整の実行方法について説明する．一般的に，形状に比べて姿勢はより安定した推定が可能であることが知られており，本手法ではこれを利用した2段階のパラメータ調整を行う．以下， $p = (p_{t-n+1}^T, \dots, p_t^T)^T$, $a = (a_{t-n+1}^T, \dots, a_t^T)^T$ とし，あるパラメータ x に関して調整範囲の制約を C_x , 調整における初期値を \hat{x} と表記する．

第一段階: 姿勢のみの調整

以上の手順により選択した n フレームの情報に対して，まず次のように姿勢のみに関する最適化を行う．

$$\min_p F_t, \quad p \in C_p \quad (4.29)$$

このとき, \hat{p} および C_p は次のように設定する.

$$\hat{p}_i = \begin{cases} p_i^{(t-1)} & (i < t) \\ p'_t & (i = t) \end{cases} \quad (4.30)$$

$$C_p = \{p \mid \hat{p}_i - \lambda_p \leq p \leq \hat{p}_i + \lambda_p\} \quad (4.31)$$

ここで $p_i^{(j)}$ はフレーム j までの観測にもとづくフレーム i の調整結果を示し, すでに一度調整が行われているフレーム $t-1$ まではこれを初期値として用いる. 新たなフレーム t に関しては, 4.4 節で述べるパーティクルフィルタによる姿勢推定の結果 p'_t を初期値として用いる. また, λ_p は定数ベクトルであり, 初期値から一定の範囲内に収まるような制約を与えている.

第二段階: 顔形状を含めた調整

次に, 形状を含めた全てのパラメータに対して最適化を行う.

$$\min_{p, a, s} F_t, \quad p \in C_p, \quad a \in C_a, \quad s \in C_s \quad (4.32)$$

まず, 姿勢パラメータに関しては初期値として第一段階の調整結果 \hat{p} を用いる.

$$\hat{p} = \hat{p} \quad (4.33)$$

C_p は式 (4.31) に従う.

また, 変形パラメータに関しては次のように \hat{a} , C_a を設定する.

$$\hat{a}_i = \begin{cases} a_i^{(t-1)} & (i < t) \\ a'_t & (i = t) \end{cases} \quad (4.34)$$

$$C_a = \begin{cases} \{a \mid \hat{a}_i - \lambda_a \leq a \leq \bar{a} + 2\sigma_a\} & \text{if } \hat{a}_i + \lambda_a > \bar{a} + 2\sigma_a \\ \{a \mid \bar{a} - 2\sigma_a \leq a \leq \hat{a}_i + \lambda_a\} & \text{if } \hat{a}_i - \lambda_a < \bar{a} - 2\sigma_a \\ \{a \mid \hat{a}_i - \lambda_a \leq a \leq \hat{a}_i + \lambda_a\} & \text{otherwise} \end{cases} \quad (4.35)$$

初期値に関しては姿勢と同様の考え方をとり, 一度調整が行われたフレームに関してはその値を, 新たなフレームに関してはパーティクルフィルタの推定値 a'_t を設定する. 制約範囲に関しても同様だが, 4.2 節でモデル構築の際に算出したパラメータの分布を元に, $\bar{a} \pm 2\sigma_a$ を超えない範囲で上限, 下限を設定する. 正規分布

を仮定したときに分布全体の95%に収まるような制限を与えることで、必要以上の変形が起こるのを防いでいる。

一方、個人差パラメータは次のように、初期値として過去 $N_s (> n)$ フレーム分の調整結果の平均を与える。

$$\hat{s} = \frac{1}{N_s} \sum_{i=t-N_s}^{t-1} s^{(i)} \quad (4.36)$$

これにより、個人差パラメータに関してはフレーム間の細かい推定変動の影響を抑え、長期的な観測にもとづく調整を行う。

$$C_s = \begin{cases} \{s \mid \hat{s}_i - \lambda_s \leq s \leq \bar{s} + 2\sigma_s\} & \text{if } \hat{s}_i + \lambda_s > \bar{s} + 2\sigma_s \\ \{s \mid \bar{s} - 2\sigma_s \leq s \leq \hat{a}_i + \lambda_s\} & \text{if } \hat{s}_i - \lambda_s < \bar{s} - 2\sigma_s \\ \{s \mid \hat{s}_i - \lambda_s \leq s \leq \hat{s}_i + \lambda_s\} & \text{otherwise} \end{cases} \quad (4.37)$$

制約範囲に関しては、変形パラメータと同様に設定する。

各パラメータの調整範囲 $\lambda_p, \lambda_a, \lambda_s$ に関しては、現在は経験的に値を設定している。個人差パラメータの調整範囲 λ_s に関しては、 λ_a に比べて相対的に小さい値を設定する。これにより、さらに個人差パラメータのフレーム間変動を抑制している。

調整に利用するフレームの選択

式(4.18)においてフレーム数 n は常に固定としているが、これは必ずしも連続した n フレームである必要はない。追跡の中で得られるフレームの中から調整に適したものを選択することで、より効果的な形状推定を行うことができる。本手法では、フレームの選択を以下のような手順で行う。

前回の調整に利用したフレームの集合を $\{f_1, f_2, \dots, f_n\}$ とすると、この中から1つを新たなフレーム f_t で置き換えることで新たな調整用集合を生成する。このとき考えられる n 通りの組み合わせ ($\{f_t, f_2, \dots, f_n\}$ から $\{f_1, \dots, f_{n-1}, f_t\}$ まで) に関して、本手法では推定姿勢角度のばらつきに注目して各集合の評価を行う。すなわち、 n 通りそれぞれの場合に与えられる姿勢初期値 (式(4.30)) に関して、顔向きに関して最も分散が大きくなる組み合わせを調整用の集合として採用する。これにより、

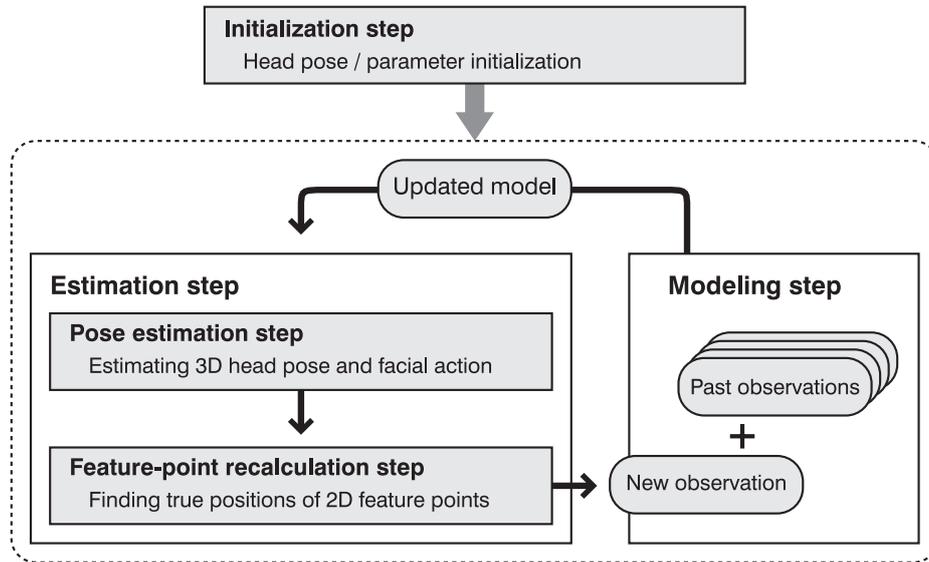


図 4.8: 姿勢推定の処理の流れ

4.4 顔形状推定を伴う頭部姿勢推定

本節では、図 4.1 における *Estimation step* の処理について説明する。4.3 節で述べた処理により並行して顔形状モデルの個人調整を行いながら、入力画像フレームに対する顔変形 a_t 及び頭部姿勢 p_t の推定を実現する。さらに本節では追跡を開始するための初期化処理についても説明し、改めて実時間頭部姿勢推定全体の流れについて整理する。処理の流れは図 4.8 のようになる。初期化ステップ (*Initialization step*) では数フレームにわたって 2 次元の顔特徴点を検出し、これに対してバンドル調整を行うことで姿勢及び顔形状の初期化を自動で行う。初期化が完了すると入力フレームに対する逐次的な推定 (*Estimation step*) に入る。*Estimation step* は、パーティクルフィルタによる姿勢、変形の推定を行う *Pose estimation step* とバンドル調整に用いるための正確な特徴点位置を再計算する *Feature-point recalculation step* により構成される。以下、各ステップの詳細について説明する。

4.4.1 顔検出と顔形状・姿勢の自動初期化

本節では *Initialization step* の詳細について説明する。処理の概要は図 4.9 に示したとおりである。ここではまず、複眼の推定システムと同様の手法で 2 次元の特徴点座標を検出する。最初にオムロン社で開発された OKAO ビジョンライブラリ

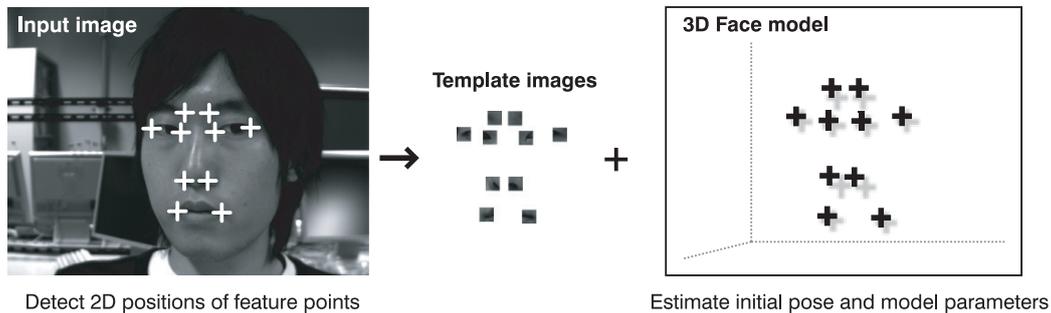


図 4.9: 顔モデルと頭部姿勢の初期化

を利用して入力画像から自動的に顔および6個の顔特徴点を検出した上で[21], 残りの $(K - 6)$ 個の特徴点について追跡に適した画像特徴を持つ点を検出する [30]. このアルゴリズムの詳細に関しては3.2.2節を参照されたい.

ただし, 4.2節で述べたように本手法では左右対称な顔形状モデルを利用しているため, ここで検出する顔特徴点もできる限り左右対称な点を抽出する必要がある. そこで本手法ではさらに, 次のような手順で対称性の判定を行う. 検出された K 個の特徴点に関して, 左右で対応する2点を結んだ $K/2$ 本の線分の傾きを算出する. この $K/2$ 個の傾きの分散が一定以上の大きさになった場合は対称性が低いと判断し, 検出失敗と見なす.

以上の処理を経て n フレーム分の顔検出に成功すると, 式 (4.18) に基づくバンドル調整を一定回数繰り返すことで頭部姿勢及びモデルパラメータの初期化を行う. この場合, 式 (4.30) のような初期推定値が得られないため, 初期化ステップにおいてはあらかじめ設定した値を各パラメータの繰り返し初期値として用いる. 第4.5節で述べる実験では, 頭部姿勢に関しては顔がほぼ画面の中央, カメラに正対するような値を, モデルパラメータに関しては平均形状に近いほぼ0の値を, それぞれ初期値として設定している.

最後に初期化時の入力画像を元に, ユーザ独自の K 個のテンプレート画像集合 T を登録する. こうして追跡を開始するのに必要な情報が得られると, 自動的に *Estimation step* の逐次処理に移行する.

4.4.2 パーティクルフィルタにおける多重線形モデルの利用

次に、本節ではパーティクルフィルタを利用した頭部姿勢と顔変形の推定について説明する．これは図 4.8 の *Pose estimation step* の処理に相当する．変形を含む姿勢推定の大まかな枠組みは 3.3.1 節と同様である．

逐次推定の対象となるのは時間的に変化するパラメータ p_t, a_t のみであり、個人差パラメータ s はその対象としない．本手法で用いる多重線形モデルにおいて、個人差パラメータ s を固定とすることで次のように変形に関する基底形状ベクトル \mathcal{M}_s が得られる．

$$\mathcal{M}_s = \tilde{\mathcal{M}} \times_{\text{shape}} s^T \quad (4.38)$$

これを用いて、式 (4.15) は変形パラメータ a_t に関する線形モデルとして表現することができる．

$$M = \bar{M} + a_t \mathcal{M}_s \quad (4.39)$$

このようにして得られるフレーム t 時点での変形モデルを用いて、各入力画像に対して $(6 + A')$ 次元状態ベクトル $x_t = (p_t^T, a_t^T)^T$ を推定する．大まかな推定の流れは 3.3.1 節と同様であり、以下、本節では個々の処理の詳細に関して説明する．

本節における仮説群 $\{(u_t^{(i)}; \pi_t^{(i)})\} (i = 1 \dots N)$ は、 $(6 + A')$ 次元状態空間内の N 個の仮説 $u_t^{(i)}$ と、各仮説に対応する重み $\pi_t^{(i)}$ で構成される．

まず、直前のフレーム $t - 1$ の仮説群 $\{(u_{t-1}^{(i)}; \pi_{t-1}^{(i)})\}$ と等速直線運動にもとづく動作モデルを用いて新たな仮説を N 個生成する．

$$u_t^{(i)} = u'_{t-1} + \tau v_{t-1} + \omega \quad (4.40)$$

このとき、 u'_{t-1} は $\{(u_{t-1}^{(i)}; \pi_{t-1}^{(i)})\}$ の中から選択された仮説であり、 τ はフレーム間の時間間隔、 v_{t-1} は直前のフレーム $t - 1$ で計算されている状態ベクトル x の速度である．

ω は仮説の拡散の性質を決めるシステム雑音であり、各要素はそれぞれ固有の分散を持つ平均が 0 のガウス雑音としている．頭部姿勢パラメータに対応する部分は 3.3.1 節と同様に、状態ベクトルの速度に応じて適応的に制御する．一方、変形パラメータに対応する部分に関してはモデル作成時に計算したパラメータの分布にもとづき、ガウス雑音の分散を $\kappa \sigma_a$ とする． κ は経験的に 0.2 に設定している．

次に，新たな仮説 $\mathbf{u}_t^{(i)}$ に対応する重み $\pi_t^{(i)}$ を決定する．これは仮説と入力画像との一致度を示す量であり，入力画像に対する仮説の一致度 $\mathcal{N}(\mathbf{u}_t^{(i)})$ をもとに，次のような関数により計算される．

$$\pi_t^{(i)} \propto \exp\left(-\frac{\left(K - \mathcal{N}(\mathbf{u}_t^{(i)})\right)^2}{2\sigma^2}\right) \exp\left(-\frac{1}{2} \sum_{b=1}^B \left(\frac{a_{t,b}^{(i)} - \bar{a}_b}{\varsigma_b}\right)^2\right) \quad (4.41)$$

第1項は $\mathcal{N}_h(\mathbf{u}_t^{(i)})$ をガウス関数により評価したもので，標準偏差 σ は経験的に 1.0 に設定している．一方，第2項は顔変形パラメータ $\mathbf{a}_t^{(i)}$ についての関数であり，この項を乗じることでモデルが過度に変形しないように拘束を与えている．このとき， $a_{t,b}^{(i)}$ ， \bar{a}_b ， ς_b はそれぞれ $\mathbf{a}_t^{(i)}$ ， $\bar{\mathbf{a}}$ ， σ_a の b 番目の要素である．以上の計算を行った上で，合計が 1 になるよう $\pi_t^{(i)}$ を正規化する．

ここで用いられる評価関数 $\mathcal{N}(\mathbf{u}_t^{(i)})$ は次のように定義される．まず，仮説 $\mathbf{u}_t^{(i)}$ が与えられたとき，形状成分にあたる $\mathbf{a}_t^{(i)}$ と式 (4.39) を用いて現在の顔形状を計算し，さらに姿勢成分 $\mathbf{p}_t^{(i)}$ によって変形後の顔モデルを移動する．その後，式 (4.17) と同じ投影関数を用いて K 個の 3 次元特徴点を入力画像 I_t に投影し，投影点周辺の画像とテンプレート集合 T 中の対応するテンプレートとの間のマッチングスコアを正規化相関により計算する．そして， K 個の投影点に関するスコアの和を $\mathcal{N}(\mathbf{u}_t^{(i)})$ の出力値とする．

最後に，仮説群 $\{(\mathbf{u}_t^{(i)}; \pi_t^{(i)})\}$ を用いて，重みが最大となる仮説の近傍に属する仮説集合の加重平均により現在の状態ベクトル \mathbf{x}_t を求める．

さらに，本手法でも 3.3.2 節と同様の half-way partitioned sampling 手法を用いて仮説の拡散・移動および評価を行う．これにより，変形を含む頭部姿勢に対しても比較的少量の仮説で効率良く確率密度関数の近似を行うことができる．

4.4.3 2次元特徴点座標の再計算

次に，本節では図 4.8 の *Feature-point recalculation step* について説明する．個人差の調整が正しく行われていない場合，4.4.2 節で推定した特徴点座標は必ずしも正確な位置にならない．そこで，本ステップでは特徴点座標の推定値 \mathbf{m}'_t をもとに真の 2 次元特徴点座標 $\hat{\mathbf{m}}_t$ を再計算し，これを式 (4.18) のバンドル調整における観測として利用することにより個人差パラメータの修正を行う．以下，計算方法の詳細について述べる．

本手法では次のような誤差関数 E_t を定義し，これを最小化することによって \hat{m}_t を決定する．

$$E_t = E_t^I + \epsilon E_t^M \quad (4.42)$$

ϵ は定数であり，本手法では経験的に 4000 に固定している．式 (4.42) は 3.2.3 節と同じ考えに基づくが，本手法では 2 次元の特徴点座標に関する計算を行うため，各々の定義は若干異なる．以下， E_t^I と E_t^M について具体的に説明する．

まず， E_t^I は次のように定義される．

$$E_t^I = \sum_{\text{ROI}} \left\{ \rho \|I_t(\hat{m}_t) - I_{t-1}(\hat{m}_{t-1})\|^2 + \|I_t(\hat{m}_t) - I_1(\hat{m}_1)\|^2 \right\} \quad (4.43)$$

ここで $\hat{m}_t \in \mathbb{R}^{2K}$ はフレーム t における真の 2 次元特徴点座標， $I_t(\hat{m}_t) \in \mathbb{R}^K$ は輝度ベクトルを示す． $I_t(\hat{m}_t)$ の k 番目の要素は入力画像 I_t における \hat{m}_t の k 番目の 2 次元座標での輝度を表している．式 (4.43) の各項は特徴点付近の注目領域における見えの誤差を表す．第 1 項は現在の画像 I_t と直前の画像 I_{t-1} との，第 2 項は現在の画像 I_t と追跡開始時の画像 I_1 との誤差を示す． ρ は第 1 項と第 2 項の間の比率を表し，現在は経験的に 4 に設定している．また，注目領域の大きさは 16×16 画素に設定している．

次に，もう一方の項 E_t^M は次のように定義される．

$$E_t^M = \|\hat{m}_t - m'_t\|^2 \quad (4.44)$$

ここで m'_t は 2 次元特徴点座標の推定値であり，4.4.2 節において推定された状態ベクトル x_t にもとづき，式 (4.17) により計算される．したがって，式 (4.44) は推定座標 m'_t と真座標 \hat{m}_t の誤差を表す．これにより，推定座標の周辺で真の特徴点座標を計算することができる．

以上の定義により得られる E_t を利用し，前フレームとの差分形状 $d\hat{m} = \hat{m}_t - \hat{m}_{t-1}$ を求めることで真座標 \hat{m}_t を計算する．

まず，3.2.3 節と同様にテイラー展開による $I_t(\hat{m}_t)$ の近似を行うことで， E_t^I は次のように $d\hat{m}$ の関数として記述できる．

$$E_t^I = \sum_{\text{ROI}} \left\{ \rho \|\hat{K}_t d\hat{m} + \Delta I\|^2 + \|\hat{K}_t d\hat{m} + \Delta I_0\|^2 \right\} \quad (4.45)$$

ただし,

$$\hat{\mathbf{K}}_t = \left. \frac{\partial \mathbf{I}_t}{\partial \hat{\mathbf{m}}_t} \right|_{\hat{\mathbf{m}}_{t-1}} \quad (4.46)$$

$$\Delta \mathbf{I} = \mathbf{I}_t(\hat{\mathbf{m}}_{t-1}) - \mathbf{I}_{t-1}(\hat{\mathbf{m}}_{t-1}) \quad (4.47)$$

$$\Delta \mathbf{I}_0 = \mathbf{I}_t(\hat{\mathbf{m}}_{t-1}) - \mathbf{I}_1(\hat{\mathbf{m}}_1) \quad (4.48)$$

とする.

また, E_t^M は次のようになる.

$$E_t^M = \|d\hat{\mathbf{m}} + \hat{\mathbf{m}}_{t-1} - \mathbf{m}'_t\|^2 \quad (4.49)$$

したがって, $\frac{\partial E_t}{\partial d\hat{\mathbf{m}}} = 0$ とおくことにより, E_t を最小にする $d\hat{\mathbf{m}}$ は次式により求められる.

$$d\hat{\mathbf{m}} = -\mathbf{D}^{-1}\mathbf{d} \quad (4.50)$$

ただし, \mathbf{D} , \mathbf{d} は次のように定義する.

$$\mathbf{D} = \sum_{\text{ROI}} \left\{ 2(\rho + 1) \hat{\mathbf{K}}_t^T \hat{\mathbf{K}}_t \right\} + \epsilon \mathbf{E}_{2K \times 2K} \quad (4.51)$$

$$\mathbf{d} = \sum_{\text{ROI}} \left\{ 2\hat{\mathbf{K}}_t^T (\rho \Delta \mathbf{I} + \Delta \mathbf{I}_0) \right\} + \epsilon (\hat{\mathbf{m}}_{t-1} - \mathbf{m}'_t) \quad (4.52)$$

式(4.50)により得られる $d\hat{\mathbf{m}}$ が一定の大きさに収束するまで以上の処理を繰り返す, 真の2次元特徴点座標 $\hat{\mathbf{m}}_t$ を得る.

ただし, 以上の処理は各特徴点が明確に見えた状態で行う必要があり, 特徴点不明瞭な状態で誤った再計算を行った場合, その後の推定にも悪影響を及ぼす. そこで本手法では, 推定された頭部姿勢がカメラの方向を向いており, かつ最大の仮説評価値が設定した閾値を上回る場合のみ, 特徴点の再計算及びバンドル調整を実行している.

4.5 評価実験

提案手法の性能を評価するために, 第3章で述べた複眼推定システムとの比較による評価実験を行った. 3.4節と同様に2台の校正済みIEEE1394カメラを設置して撮影した画像列に対して, まず両方のカメラを用いてユーザの頭部姿勢及び

顔形状を測定する．その後，提案手法により一方のカメラのみを用いて推定した結果と比較することにより，提案手法の精度を評価する．

本実験では，Intel Core 2 Duo E6700 を搭載した汎用 PC を 1 台使用した．搭載メモリは 3.0GB であり，OS は Windows XP を利用している．入力画像は 640×480 画素であり，画像テンプレートの大きさは 16×16 画素，パーティクルフィルタの仮説の総数は 750 であった．バンドル調整に用いるフレーム数は $n = 5$ ，個人差パラメータの平均を算出するためのフレーム数は $N = 15$ とした．LM アルゴリズムにおける繰り返しの回数は，第一段階と第二段階それぞれ 5 回，合計 10 回に制限した．初期化ステップのみ，各 10 回，合計 20 回の調整を行った．このとき，初期化ステップの処理時間は約 70[ms] となった．また，追跡中の処理時間はバンドル調整を行った場合で平均 30[ms]，顔向きなどの条件により調整を行わない場合で平均 7[ms] となり，本システムは毎秒 30 フレームで動作した．

実験では， $S = 11$ 人の人物から獲得した， $A = 10$ 通り（図 4.4 の二つの変形についてそれぞれ 5 個）の変形を元に構築したモデルを用いた． $3K \times 22 \times 10$ のデータテンソルを元に，個人差パラメータ 10 次元，変形パラメータ 5 次元の顔形状モデルを作成した．

A. モデルに被験者自身のデータが含まれる場合

まず，被験者自身の情報がモデルに含まれる場合について実験を行った．すなわち，ここで実験対象となるユーザは，モデルの学習対象である 11 人の中に含まれている．

入力として用いたのは 60 秒（1800 フレーム）分の画像列であり，被験者は顔の変形を含む頭部運動を行っている．最初に，2 台のカメラにより撮影した画像列に対して複眼システムによる姿勢推定を行った．本実験では一度姿勢推定を行った後にモデルを保持したまま再度推定を行い，2 度目の推定結果を複眼システムの推定値として利用した．これは，CCIPCA により逐次的な変形モデル学習を行う場合，追跡の初期段階では十分変形に対応できない場合があるためである．次に，一方のカメラ画像のみを用いて提案手法による単眼推定を行い，複眼推定の結果に対する誤差を評価した．

図 4.10 に，頭部姿勢推定の結果画像を示す．左列が複眼による推定結果，中央が単眼による推定結果であり，推定された頭部姿勢及び特徴点座標が描画されて

	x	y	z		roll	pitch	yaw
平均 [mm]	2.60	3.12	9.56	平均 [degree]	0.43	2.45	2.33
標準偏差 [mm]	2.24	1.73	9.69	標準偏差 [degree]	0.31	2.08	1.95

表 4.1: 頭部姿勢の推定誤差 (実験 A)

いる。また、右列は z 軸 (奥行き) 方向の推定誤差を確認するために二つの推定結果を異なる視点から描画した図である。赤い+が複眼、青い+が単眼の推定結果を示す。このように、顔変形や奥行き方向の大幅な変動を伴う姿勢に対しても安定した追跡が可能である。

さらに、二つの頭部姿勢推定結果の詳細なグラフを図 4.12 に示す。赤い細線が複眼推定、青い太線が単眼推定の結果を表わす。 x は水平方向、 y は鉛直方向、 z は深さ方向の運動、 $roll$ は z 軸回り、 yaw は y 軸回り、 $pitch$ は x 軸回りの回転である。このとき、複眼推定に対する単眼推定の誤差は表 4.1 に示すとおりである。 x 、 y 、 $roll$ はほぼ複眼での推定結果推定に一致しており、比較的単眼での推定が難しい z 及び $pitch$ 、 yaw の推定結果に関しても高い精度での推定を実現していることがわかる。

また、頭部座標系における各顔特徴点の3次元座標の平均誤差を図 4.1 に示す。複眼推定とは異なり、単眼推定では左右対称のモデルを用いるため、二つの推定結果は必ずしも厳密に一致するとは限らないが、変形が起こった場合も大きな誤差を生じさせることなく、フレーム間平均誤差 5.9[mm] での形状推定を実現している。

B. モデルに被験者自身のデータが含まれない場合

次に、同じ画像列を用いてモデルに被験者自身の情報が含まれない場合について実験を行った。すなわち、前述のモデル構築用データから被験者自身のサンプルを除き、 $S = 10$ 人、 $A = 10$ 通りの変形を元に作成したモデルを元に推定を行った。個人差パラメータの次元数は変わらず、個人差パラメータ 15 次元、変形パラメータ 5 次元とした。そのほかの実験条件も全て同じである。

複眼の推定値に対して実験 A と実験 B の推定結果を比較したグラフを図 4.13 に示す。このように、学習済みのユーザに比べて全体に推定誤差が大きくなるもの

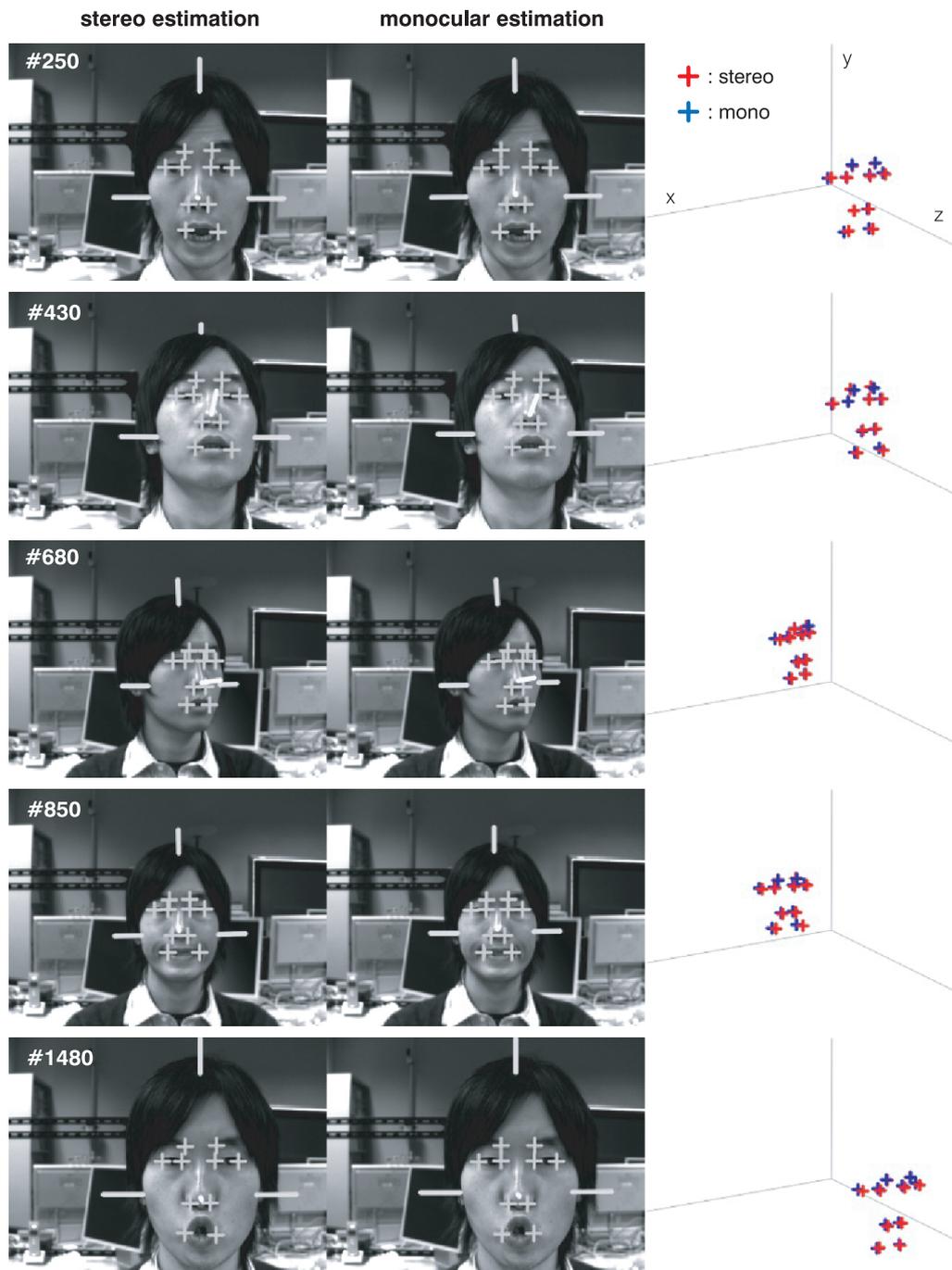


図 4.10: 頭部姿勢推定の結果画像 (実験 A)

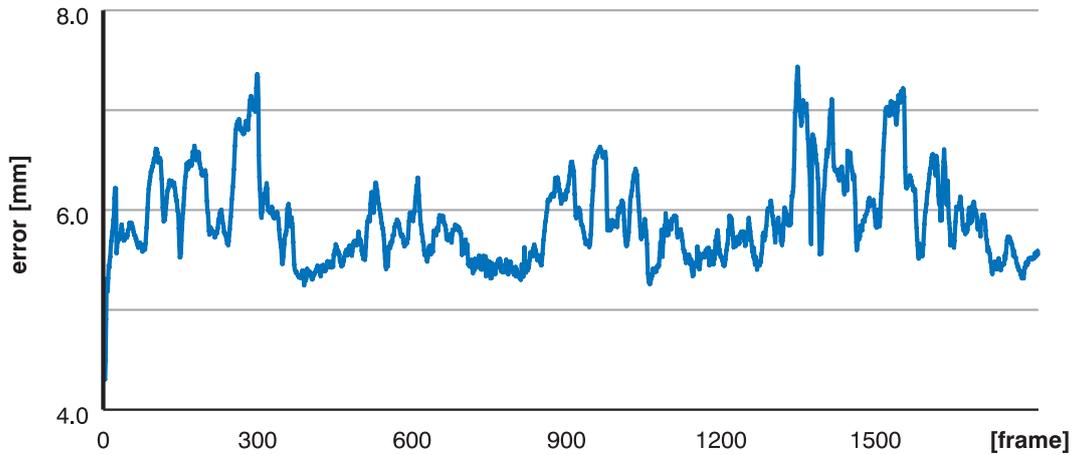


図 4.11: 推定された顔特徴点座標 (頭部座標系) の平均誤差 (実験 A)

	x	y	z		roll	pitch	yaw
平均 [mm]	3.73	1.92	17.85	平均 [degree]	0.43	3.07	2.49
標準偏差 [mm]	2.57	1.47	9.37	標準偏差 [degree]	0.32	2.31	1.96

表 4.2: 頭部姿勢の推定誤差 (実験 B)

の、未知のユーザに対しても大きな精度劣化を生じさせることなく姿勢推定が可能である。このとき、複眼推定との誤差は表 4.2 のようになった。特に z に顕著であるが、表 4.1 と比べて誤差平均が大きい結果になっており、モデル自身が持つ近似誤差が姿勢推定全体に影響していると考えられる。

C. 初期化のみで逐次的なバンドル調整を行わない場合

最後に、本手法で用いる逐次的な個人差調整の効果を検証するために、毎フレームのバンドル調整を行わない場合の推定結果との比較を行った。Estimation step では個人差パラメータを固定としてパーティクルフィルタによる姿勢・変形推定のみを実行し、時系列推定に移行するために必要な初期化ステップでのみバンドル調整を行った。モデルは実験 A と同じく、被験者自身の情報が含まれるものを用いた。

この場合の推定結果を実験 A の結果と比較したグラフを図 4.14 に示す。逐次的バンドル調整を行わない場合、姿勢推定の安定性が大きく損なわれ、これは図 4.14

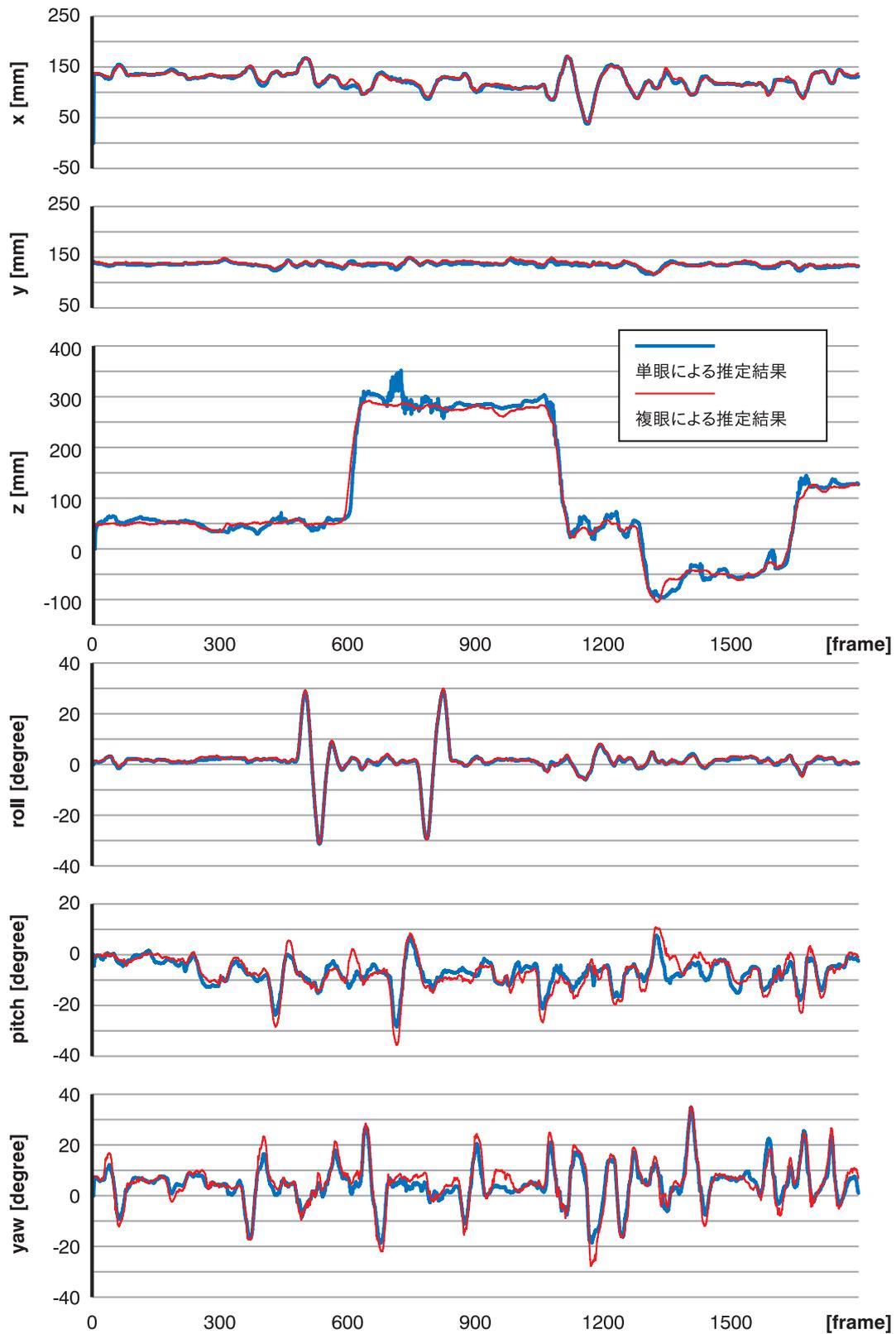


図 4.12: 頭部姿勢推定の結果 (実験 A)

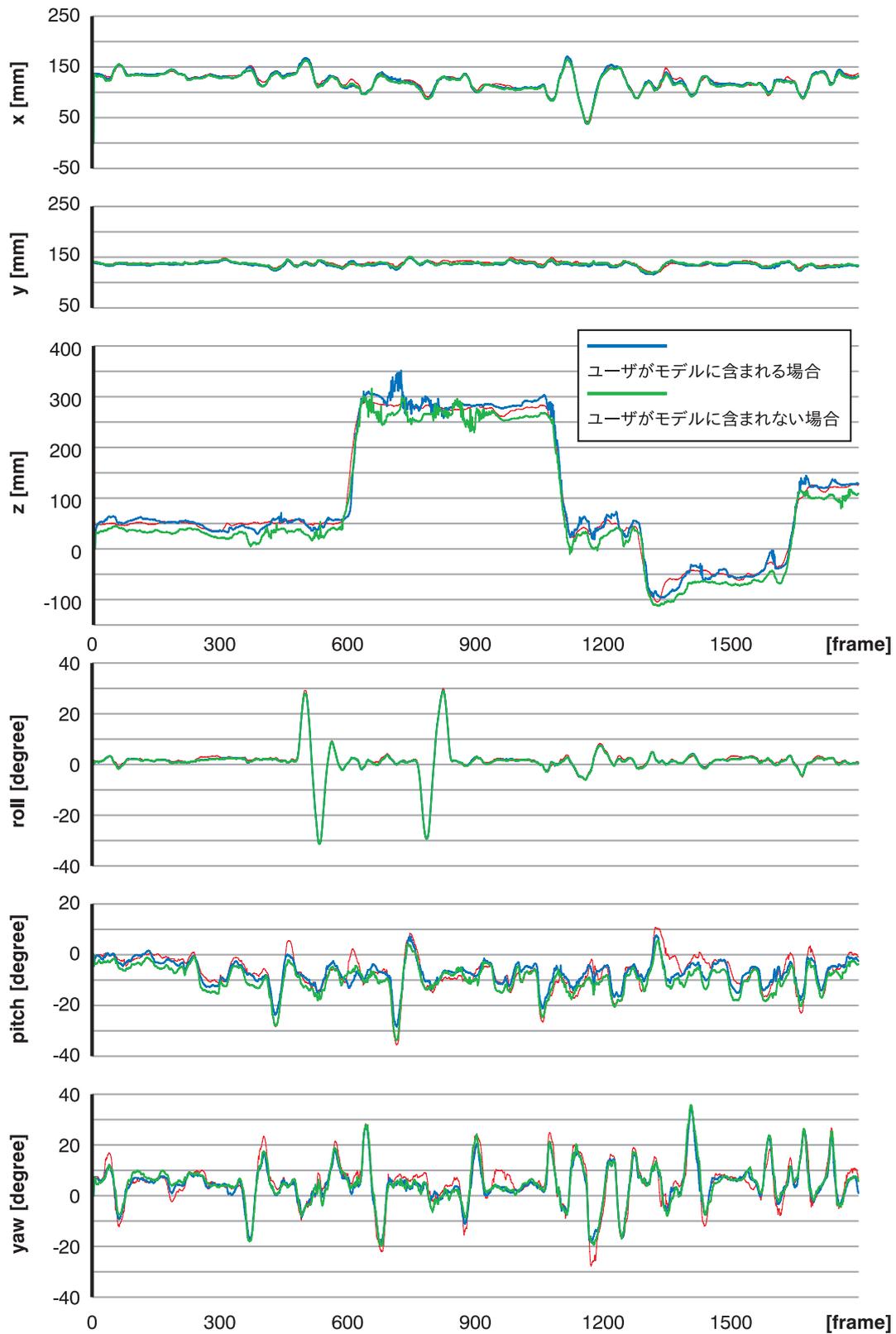


図 4.13: 頭部姿勢推定の結果 (実験 B)

	x	y	z		roll	pitch	yaw
平均 [mm]	2.71	3.43	11.38	平均 [degree]	0.43	2.52	2.26
標準偏差 [mm]	2.11	2.03	11.43	標準偏差 [degree]	0.32	2.03	2.02

表 4.3: 頭部姿勢の推定誤差 (実験 C)

において特に z の推定に顕著に表れている。複眼推定との誤差は表 4.3 のようになり、 z の推定に関しては平均だけでなく標準偏差も増加している。

このように、本手法で用いた逐次的なバンドル調整の枠組みは、推定精度及び安定性の向上に寄与していることがわかる。

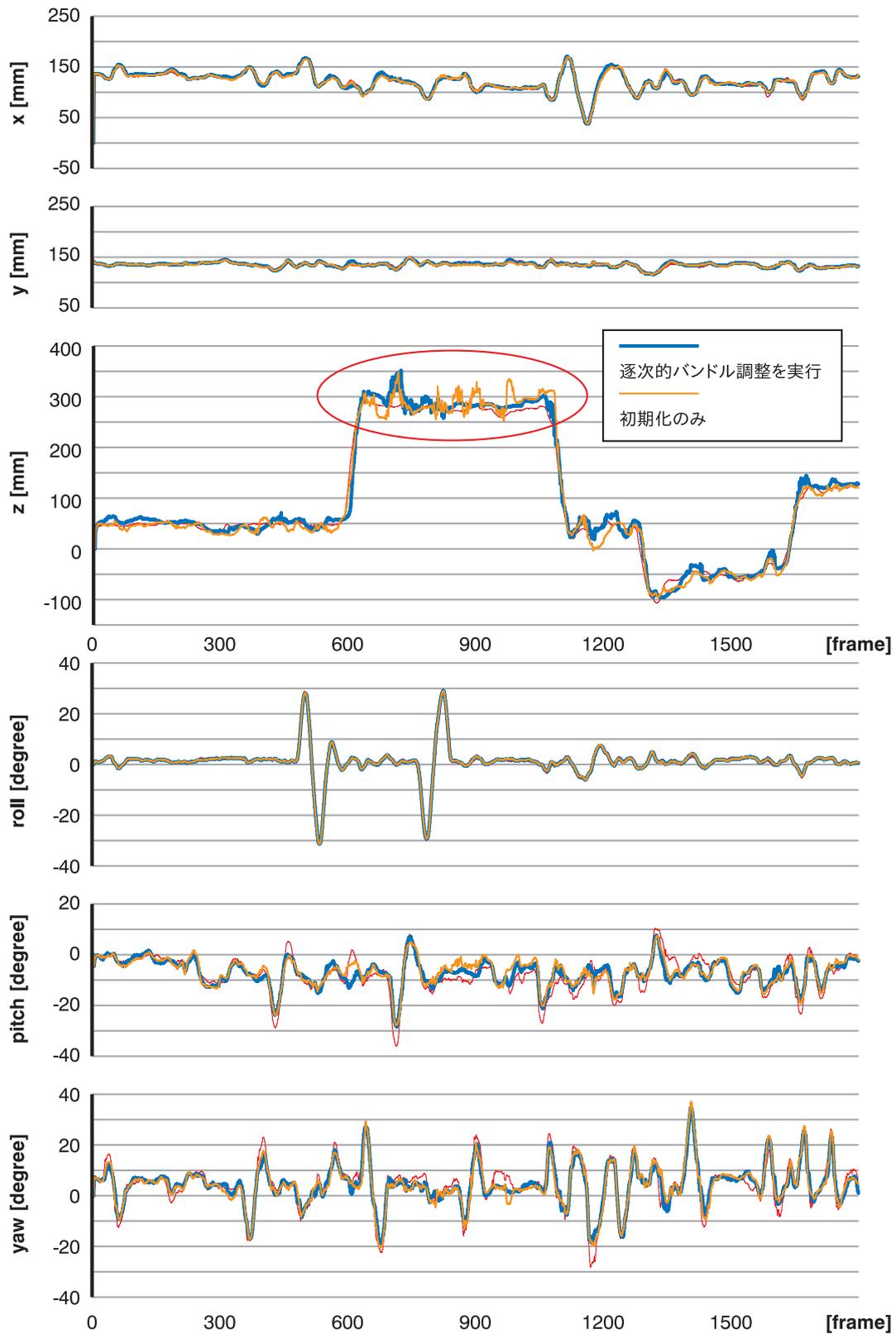


図 4.14: 頭部姿勢推定の結果 (実験 C)

第5章 結論

5.1 本研究のまとめ

本論文では、HCI 技術への柔軟な応用が可能となる、以下の2点を同時に実現する実時間頭部姿勢推定手法を提案した。

- 不特定多数のユーザに対して自動的に推定を開始できる
- 発話や表情変化に伴う顔変形が起こった場合でも、安定して推定を継続できる

顔変形を伴う実時間頭部姿勢推定を不特定多数のユーザに対して実現することは、現在でも難しい課題になっている。これを解決するために、本論文では二つの異なるアプローチに基づく手法を提案した。

顔変形モデルの自動構築を伴う頭部姿勢推定

まず第3章では1つめの手法として、3次元頭部姿勢の実時間推定と同時に顔変形モデルを自動的に構築するための手法を提案した。複眼カメラを用いて取得したユーザ頭部の3次元形状を元に、CCIPCA アルゴリズムを利用した増分的な変形モデルの構築を行う。本研究の主な貢献は以下の2点にまとめられる。

1. 姿勢推定と並行して頭部変形モデルを連続的に更新する。
2. 頭部姿勢と顔形状の推定性能を逐次的に向上させる。

本研究は、頭部変形モデルの構築と頭部姿勢の推定を実時間で同時に実行することを試みた最初の研究例であると考えられる。

本手法の特徴として、それまでの追跡中に起こった変形に対してはモデルを拡張して対応することが可能である、という点が挙げられる。したがって、ユーザ

毎の変形の特徴を効率良く捉えた柔軟な追跡が可能になる。また、複数のカメラを用いることにより、追跡の精度と安定性も向上する。姿勢推定に一定の精度が要求されるアプリケーションや環境設置型のシステムにおける利用などを想定した場合、大変有効な手法であると言える。

パラメータ分離モデルを用いた単眼カメラによる頭部姿勢推定

一方、第4章では2つめの手法として、事前に構築した顔形状モデルを元に単眼カメラのみで顔変形及び頭部姿勢の推定を行う手法を提案した。顔形状の個人内変動（変形パラメータ）と個人間変動（個人差パラメータ）を分離して表現したモデルに対して、パーティクルフィルタとバンドル調整を各パラメータの性質に合わせて利用する。これにより、単眼カメラでのシステム構築が可能になる。本研究の主な貢献は次のようになる。

1. 個人差調整を推定と並行して同時に行うことで、推定性能を逐次的に向上させる。
2. パラメータ分離モデルを用いた変形・姿勢推定を実時間システムの中で実現する。

推定精度に関しても、複眼システムにも遜色のない性能を実現している。

統計的性質に基づく多重線形モデルの利用、モデル空間内での時系列トラッキング、そして複数フレームに対する全パラメータの最適化と、従来手法と比べてもより妥当な形で個々の技術を統合した手法であると言える。単眼カメラでのシステム構築が可能であることは本手法の大きな特徴であり、複眼のシステムと比較して大幅に設置コストの低い、個人向けのアプリケーションやモバイル機器での利用に適した手法になっている。

5.2 今後の課題

複眼システムに関して

前者の複眼システムに関しては、追跡の中で初めて見られる変形に対する表現力は乏しくなるという欠点がある、特に追跡開始直後、モデルの学習が十分行わ

れていない場合には，極端に大きい変形には対応できない場合も考えられる．

また，モデル構築の正確さには特徴点位置の再計算精度も大きく影響するが，特徴点の変動から変形による成分だけを厳密に抽出するのは非常に難しい問題である．姿勢変動と変形の厳密な切り離しを実現することで，モデルのみならず頭部姿勢に関してもより正確な評価が可能になると言える．

今後の課題として，こうした問題に対応することでさらなる追跡精度の向上を目指すことが挙げられる．

単眼システムに関して

後者の単眼システムに関しては，まずは未知ユーザに対する推定性能をより厳密に評価することが課題として挙げられる．現在はモデル構築時の学習人数が限られた状態での実験を行っているため，今後はさらに学習人数を増やしてより正確な評価を行う必要がある．

推定精度については，モデル構築手法そのものを見直すことで改善できると考えられる．現在は個人性と表情というあくまでも意味上の分離によりデータを収集とモデル構築を行っているが，姿勢推定精度の観点から見ると，姿勢に影響のある変形と影響のない変形を分離することも重要である．変形パラメータが姿勢変動と同じ動きを含む場合，その正確な推定は非常に困難となる．現在のモデルにおいてこれは縦方向の回転（pitch）に顕著であり，推定精度悪化の原因になっている．

さらに，横向き（yaw）方向の追跡可能範囲や，奥行き（z）方向の追従性に関しては，そもそも単眼で推定を行う上で避けられない問題になっている．今後はこうした問題にも対処しながら，より実用に適したシステム構築を目指す予定である．

より詳細な顔情報解析への取り組み

両手法共通の今後の課題としては，感情推定などへの応用を目的としたモデルの拡張が挙げられる．現在のモデルは必要最低限の特徴点で構成されており，目・口の開閉や眉の形状が捉えられないことは，表情の解析を行う上で大きなネックになっている．今後はメッシュモデルの利用なども視野に入れながら，より詳細な

顔形状モデルを利用した推定を実現する。さらに、瞬き検出や視線推定など、顔に関する発展的なセンシング技術を統合することで、総合的なセンシングシステムの構築を目標とする。

システムの特性を生かしたアプリケーションの提案

最後に、本手法の特性を生かした応用アプリケーションの提案が課題として挙げられる。不特定多数のユーザが利用する公共ディスプレイでの利用や、ウェブカメラを利用したデスクトップ環境での個人向けアプリケーションなどを通して、実際の具体的応用例と共に本手法の有効性を示していきたいと考えている。

謝辞

本研究を遂行するにあたっては、非常に多くの方々のご指導とご協力を賜りました。この場を借りて、皆様に心から感謝の意を表します。

修士課程の2年間に渡り日頃から熱心にご指導頂き、本研究を遂行する機会を与えて下さいました佐藤洋一助教授に、深く御礼申し上げます。

本研究の基礎となるシステムのコードを提供して頂き、数多くの貴重なご助言を頂きました岡兼司さんに深く感謝いたします。

岡部孝弘助手、佐藤いまり助手をはじめとして、研究生活を様々な面から支えて頂いた佐藤研究室の皆様に深くお礼申し上げます。

そして最後に、日々心の支えとなってくれた全ての友人と、学生生活を支えてくれた母に心から感謝いたします。

ありがとうございました。

2007年2月2日

菅野 裕介

参考文献

- [1] H. Aanaes and F. Kahl: Estimation of deformable structure and motion, *Proc. Workshop on Vision and Modelling of Dynamic Scenes, ECCV'02*, (2002).
- [2] J. Ahlberg: CANDIDE-3 – an updated parameterized face, *Technical Report LiTH-ISY-R-2326*, Dept. of Electrical Engineering, Linköping University (2001).
- [3] B. Bascle and A. Blake: Separability of pose and expression in facial tracking and animation, *Proc. IEEE Int. Conf. Computer Vision*, pp. 323–328 (1998).
- [4] S. Basu, I. Essa and A. Pentland: Motion Regularization for Model-based Head Tracking, *Proc. Int. Conf. Pattern Recognition*, Vol. 3, pp. 611–616 (1996).
- [5] C. Bregler, A. Hertzmann and H. Biermann: Recovering non-rigid 3d shape from image streams, *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, Vol. 2, pp. 690–696 (2000).
- [6] T. F. Cootes, G. J. Edwards, and C. J. Taylor: Active Appearance Models, *Proc. European Conf. on Computer Vision (ECCV 1998)*, pp. 484–498 (1998).
- [7] D. DeCarlo and D. Metaxas: Optical flow constraints on deformable models with applications to face tracking, *Int. J. Computer Vision*, Vol. 38, No. 2, pp. 99–127 (2000).
- [8] D. DeCarlo and D. Metaxas: Adjusting shape parameters using model-based optical flow residuals, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 24, No. 6, pp. 814–823 (2002).
- [9] A. Del Bue, X. Llado and L. Agapito: Non-rigid structure from motion using non-parametric tracking and non-linear optimization, *Proc. IEEE Workshop on Articulated and Non-Rigid Motion*, Vol. 1, p. 8 (2004).

- [10] A. Del Bue, X. Llado and L. Agapito: Non-rigid metric shape and motion recovery from uncalibrated images using priors, *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, Vol. 1, pp. 1191–1198 (2006).
- [11] M. Dimitrijevic, S. Ilic and P. Fua: Accurate face models from uncalibrated and ill-lit video sequences, *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, Vol. 2, pp. 1034–1041 (2004).
- [12] F. Dornaika and F. Davoine: Simultaneous facial action tracking and expression recognition using a particle filter, *Proc. IEEE Int. Conf. Computer Vision*, Vol. 2, pp. 1733–1738 (2005).
- [13] F. Dornaika and A. D. Sappa: Rigid and Non-rigid Face Motion Tracking by Aligning Texture Maps and Stereo-Based 3D Models, *Proc. Advanced Concepts for Intelligent Vision Systems (ACIVS 2006)*, pp. 675–686 (2006).
- [14] A. Gee and R. Cipolla: Fast visual tracking by temporal consensus, *Image and Vision Computing*, Vol.14, pp.105–114 (1996).
- [15] S. B. Gokturk, J. Y. Bouguet and R. Grzeszczuk: A data-driven model for monocular face tracking, *Proc. IEEE Int. Conf. Computer Vision*, Vol. 2, pp. 701–708 (2001).
- [16] R. Gross, I. Matthews and S. Baker: Generic vs. person specific active appearance models, *Image and Vision Computing*, Vol. 23, No. 11, pp. 1080–1093 (2005).
- [17] G. D. Hager and P. N. Belhumeur: Efficient region tracking with parametric models of geometry and illumination, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 20, No. 10, pp. 1025–1039 (1998).
- [18] Isard, M. and Blake, A.: Condensation– conditional density propagation for visual tracking, *Int. J. Computer Vision*, Vol.29, No.1, pp.5–28 (1998).
- [19] C. Kanzow, N. Yamashita and M. Fukushima: Levenberg-Marquardt methods for constrained nonlinear equations with strong local convergence properties, *J. Computational and Applied Mathematics*, Vol. 172, pp. 375–397 (2004).

- [20] M. La Cascia, and S. Sclaroff and V. Athitsos: Fast, reliable head tracking under varying illumination: an approach based on registration of texture-mapped 3D models, *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 22, No. 4, pp. 322–336 (2000).
- [21] S. Lao, T. Kozuru, T. Okamoto, T. Yamashita, N. Tabata and M. Kawade: A fast 360-degree rotation invariant face detection system, *Demo session of IEEE Int. Conf. Computer Vision* (2003).
- [22] L. De Lathauwer, B. De Moor and J. Vandewalle: A multilinear singular value decomposition, *SIAM Journal on Matrix Analysis and Applications*, Vol. 21, No. 4, pp. 1253–1278 (2000).
- [23] L. De Lathauwer, B. De Moor and J. Vandewalle: On the best rank-1 and rank- (R_1, R_2, \dots, R_N) approximation of higher-order tensors, *SIAM Journal on Matrix Analysis and Applications*, Vol. 21, No. 4, pp. 1324–1342 (2000).
- [24] M. I. A. Lourakis: levmar: Levenberg-Marquardt nonlinear least squares algorithms in C/C++, <http://www.ics.forth.gr/~lourakis/levmar/> (2004).
- [25] J. MacCormick and M. Isard: Partitioned sampling, articulated objects, and interface-quality hand tracking, *Proc. European Conf. Computer Vision*, Vol. 2, pp. 3–19 (2000).
- [26] Y. Matsumoto and A. Zelinsky: An algorithm for real-time stereo vision implementation of head pose and gaze direction measurement, *Proc. IEEE Int. Conf. Automatic Face and Gesture Recognition*, pp. 499–504 (2000).
- [27] I. Matthews and S. Baker: Active appearance models revisited, *Int. J. Computer Vision*, Vol. 60, No. 2, pp. 135–164 (2004).
- [28] E. Munoz, J. M. Buenaposada, L. Baumela: Efficient model-based 3D tracking of deformable objects, *Proc. IEEE Int. Conf. Computer Vision*, pp. 877–882 (2005).

- [29] Y. Shan, Z. Liu and Z. Zhang: Model-based bundle adjustment with application to face modeling, *Proc. IEEE Int. Conf. Computer Vision*, pp. 644–651 (2001).
- [30] Tomasi, C. and Kanade, T.: Shape and motion from image streams: a factorization method–3, detection and tracking of point features, *Technical Report CMU-CS-91-132*, (1991).
- [31] L. Vacchetti, V. Lepetit and P. Fua: Stable real-time 3D tracking using online and offline information, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 26, No. 10, pp. 1380–1384 (2004).
- [32] M. A. O. Vasilescu and D. Terzopoulos: Multilinear analysis of image ensembles: TensorFaces, *Proc. European Conf. on Computer Vision (ECCV'02)*, pp. 447–460 (2002).
- [33] M. A. O. Vasilescu and D. Terzopoulos: TensorTextures: multilinear image-based rendering, *ACM Transactions on Graphics (Proc. ACM SIGGRAPH 2004)*, pp. 336–342 (2004).
- [34] D. Vlasic, M. Brand, H. Pfister and J. Popovic: Face transfer with multilinear models, *ACM Transactions on Graphics (Proc. ACM SIGGRAPH 2005)*, Vol. 24, No. 3, pp. 426–433 (2005).
- [35] J. Weng, Y. Zhang and W. S. Hwang: Candid Covariance-Free Incremental Principal Component Analysis, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.25, No.8, pp.1034–1040 (2003).
- [36] J. Xiao, S. Baker, I. Matthews, and T. Kanade: Real-Time Combined 2D+3D Active Appearance Models, *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, Vol. 2, pp. 535–542 (2004).
- [37] J. Xiao, B. Georgescu, X. Zhou, D. Comaniciu and T. Kanade: Simultaneous Registration and Modeling of Deformable Shapes, *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, pp. 2429–2436 (2006).

- [38] L. Xin, Q. Wang, J. Tao, X. Tang, T. Tan and H. Shum: Automatic 3D face modeling from video, *Proc. IEEE Int. Conf. Computer Vision*, Vol. 2, pp. 1193–1199 (2005).
- [39] Z. Zhang: Flexible camera calibration by viewing a plane from unknown orientation, *Proc. IEEE Int. Conf. Computer Vision (ICCV '99)*, pp. 666–673 (1999).
- [40] Y. Zhang and J. Weng: Convergence Analysis of Complementary Candid Incremental Principal Component Analysis, *Technical Report MSU-CSE-01-23*, Dept. of Computer Science and Eng., Michigan State Univ. (2001).
- [41] Z. Zhang and Y. Shan: Incremental motion estimation through modified bundle adjustment, *Proc. IEEE Int. Conf. Image Processing*, Vol. 2, pp. 343–346 (2003).
- [42] Z. Zhu and Q. Ji: Robust Real-Time Face Pose and Facial Expression Recovery, *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, pp. 681–688 (2006).
- [43] 岡兼司, 佐藤洋一, 中西泰人, 小池英樹: 適応的拡散制御を伴うパーティクルフィルタを用いた頭部姿勢推定システム, *電子情報通信学会論文誌 D-II*, Vol. J88-D-II, No. 8, pp. 1601–1613 (2005).

発表文献

1. 菅野裕介, 佐藤洋一: 表情変動を許容した実時間頭部姿勢推定のための個人間および個人内変動に対する顔形状推定, 情報処理学会コンピュータビジョンとイメージメディア研究会, 2006-CVIM-156-21, pp. 179-186 (2006).
2. 岡兼司, 菅野裕介, 佐藤洋一: 頭部変形モデルの自動構築を伴う実時間頭部姿勢推定, 情報処理学会論文誌 Vol. 47 No. SIG 10 (CVIM 15) pp. 185-194 (2006).