Vol.2016-CVIM-202 No.44
2016/5/13

奥行き情報を利用した顔・髪の検出と追跡に関する研究

鈴木 一正[1,a)]　呉 海元[1,b)]

概要：本論文では，奥行き情報を利用した顔・髪領域やこれらを合わせた頭部領域の検出・追跡システムを提案する．ステレオ画像から得られるスパースな奥行き情報を利用した識別回数の削減によって顔検出の高速化を行い，検出精度を向上させながら単眼カメラよりも高速なビデオレートでの顔検出を実現する．顔追跡には色弁別度追跡法を用い，顔識別器により顔だけを正確に検出し，検出された顔領域内外の情報を基に環境に応じた追跡モデルを自動的に構築する．また，対象までの奥行き情報を用いた制約により顔追跡の安定化を実現する．髪追跡では，奥行き情報を活用した 6D K-means Tracker を提案し，3 次元化された位置情報によって前景と背景の特徴同士の分離性能を高めることで追跡の安定化を行い，色彩情報の乏しく形状変化が大きい髪領域の追跡を実現する．

Face and Hair Detection and Tracking Using Depth Information

Kazumasa Suzuki[1,a)]　Haiyuan Wu[1,b)]

1. はじめに

1.1 研究の背景・目的

現在，人々の生活は機械に囲まれており，自動車や家電製品など日常的に利用するものから，インフラや産業ロボットなど社会を支えるものまで，あらゆる分野において人と機械は密接な関係を持っている．コンピュータの発展にともなって大量の情報処理ができるようになり，知能を持った機械が人を支援するようになってきた．機械が人を補助するためには，人の状態や周囲の環境を認識する必要がある．

本研究では，ステレオカメラや Kinect から取得された画像中において，人の状態を認識する上で重要な顔・髪領域からなる頭部領域を決定することを目的とし，撮影された画像から顔を高速かつ高精度に検出する方法を提案し，検出された顔領域に基づいて，ビデオレートで顔・髪・頭部領域を追跡するシステムを構築する．

コンピュータビジョンの分野では，顔画像処理が 1970 年代頃より始まり，80 年代後半から 90 年代には盛んに研究されるようになった．90 年代後半にはセキュリティ応用を目的とした，より高速で安定した顔検出・認識手法が提案され，商品化されている．顔画像に関する研究は，顔認識，表情認識，年齢推定，男女判定など多岐にわたる．顔の位置や姿勢，表情などの情報をコンピュータで読み取ることができれば，監視カメラやマシンインターフェース，ロボットとの対話など様々な応用システムが考えられる．これらの顔画像処理を行うためには，まず画像中から顔の領域を見つける必要がある．また，継続的に顔領域を得るために，顔検出・追跡は重要な処理であり，実用的なシステムを構築するために高速かつ安定な手法が求められる．一方，髪の色やスタイルは千差万別でありながら，画像内の対応画素には色情報やテクスチャ情報が少なく，モデル化や安定な特徴を抽出することが難しい．そのため，髪領域検出に関わる研究の報告例は少なく，ビデオレートでの追跡を行っている例は見当たらない．しかし，髪領域は重要な個人特徴であり，髪型を個人識別のための特徴として用いたり，顔と髪領域の関係を用いた姿勢推定など様々な応用アプリケーションで利用することが考えられる．また，顔領域だけの追跡では頭部とカメラの相対的な姿勢変化によって継続した追跡ができない場合でも，頭部の 360 度という全方位にある髪領域の追跡を行うことで頭部の姿

[1] 和歌山大学
Wakayama Uniersity
[a)] suzuki.kazumasa@g.wakayama-u.jp
[b)] wuhy@center.wakayama-u.ac.jp

勢に関わらない安定した追跡が可能となる．そこで本研究では，顔と髪をそれぞれ単独対象とした検出・追跡ができるシステムを構築すると同時に，顔と髪からなる頭部領域を対象とした検出・追跡システムを構築する．

1.2 従来手法

1990 年代後半には，コンピュータの処理速度の向上にともなって，大量のデータと高次元の特徴量を処理できるようになってきたため，機械学習による顔検出手法が提案されるようになった．これらの手法では，顔と非顔にラベリングされた大量の画像を学習することで自動的に顔モデルを作成できるため，汎用性の高い識別器を構築することができる．中でも Viola ら [1] が提案している AdaBoost で学習した強識別器をカスケードに接続した手法は，ビデオレートでの顔検出の可能性が初めて示され，その高速性と，汎化性能の高さから，単に学術的研究としてだけでなく，デジタルカメラやプリンタ，監視機器など工業製品への組み込みが行われるなど実用的側面からも注目を集めている．

顔検出の研究は識別器に関するものが多いが，中には色情報やフィルタを用いて大部分の背景を除去することで識別回数を削減し，既存の識別手法を高速化しているものもある．勞ら [2] は肌色を顔検出ではなく，明らかに顔が含まれない領域を除去する前処理として使うことで，顔検出の高速化を行っている．Shaick ら [3] は顔モデルを用いたフィルタ処理により画像をセグメンテーションし，背景領域を大幅に削減している．このような，前処理で背景領域を削減する手法は，誤って顔領域も削減してしまう可能性がある．また，FPGA（field-programmable gate array）[4] や GPU[5] などのハードウェアを用いた並列処理によって高速化している手法もある．

近年，奥行き情報を利用した顔検出や認識の研究 [6], [7] も行われており，Microsoft 社から Kinect 用の顔追跡ライブラリ（Face Tracking SDK[8]）も提供されている．R.I. Hg ら [6] は，Kinect を用いて顔の RGB-D データベースを構築し，顔検出システムを構築している．T. Huynh ら [7] は，Kinect から得られた距離情報に基づいて，顔表面の法線ベクトルを表現する LBP 型の記述子を提案し，性別の識別を行っている．これらは，奥行き情報を利用した検出精度や認識精度の改善を行っている．

一方，髪の色やスタイルは千差万別であり，モデル化が難しい．また，画像内の対応画素の色やテクスチャの情報が乏しいため，検出・識別に有効かつ安定な特徴を抽出することが難しい．そのため，現状のコンピュータビジョンの研究分野では，顔と比べると髪領域の自動検出に関する研究は少なく，髪領域をビデオレートで追跡する例は見当たらない．

Y.Yacoob ら [9] は，顔検出と目検出で得られた幾何学的な位置関係から肌の色と髪の色を取得してモデル化し，髪色モデルと類似した画素を連結することで髪領域を検出している．K.Lee ら [10] は，色と画素位置の情報を用いた，Graph-Cut や belief propagation による顔・髪・背景領域のセグメンテーションを行っている．P.Julian ら [11] は単純な統計髪型モデルを用いて頭部の髪領域を表現し，active contour model と active shape model の組み合わせによって髪の初期領域を検出している．髪の色とテクスチャをこの初期領域から appearance パラメータとして学習し、得られたパラメータで髪の領域が分割される．これらの手法では，髪の形状に基づいたモデルを定義しているため正面顔でのセグメンテーションにのみ対応しており，テクスチャ情報を得るために高解像度の画像が必要となる．また，髪と同色の背景が隣接しないような比較的単純な背景下での検出を行っている．

髪領域の検出に関する研究の多くは，色や形状により定義されたモデルをあらかじめ用意しており，頭部姿勢の変化を伴う連続的な髪領域の検出を行うことはできない．また，リアルタイム性を求められる動画像へ適用する場合，処理コストの低い方法が求められる．

1.3 提案手法の概要

顔画像処理において，画像中の顔位置を特定することは最も基礎的で重要な処理であり，また，髪の色やスタイルには重要な個人特徴が含まれている．そこで本論文では，画像中から顔・髪・頭部領域の決定を高速かつ安定に行うことを目的とし，毎秒 30 フレームのビデオレートで撮影された動画像において顔や髪領域を自動的に検出・追跡するシステムを提案する（図 1）．提案システムは，識別器による顔検出と色情報を用いた追跡手法の組み合わせで構成され，顔検出で得られた領域から追跡に必要な情報を取得することによって，その時の環境に応じたターゲット・背景モデルを構築し，自動的に追跡が開始できるシステムとなっている．本論文で提案する手法は以下の 3 点であり，検出や追跡において奥行き情報を利用することで処理の高速化や安定化を実現する．

提案手法 1　奥行き情報を用いた顔検出の高速化

提案手法 2　顔検出と色弁別度追跡法を組み合わせた顔追跡システム

提案手法 3　顔検出と 6D K-means Tracker を組み合わせた髪追跡システム

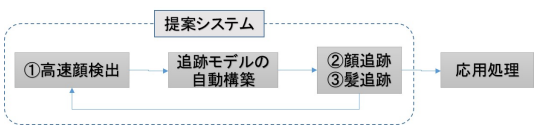


図 1　提案システムの全体像

2. 奥行き情報を用いた顔検出の高速化

一枚の画像から顔の位置を検出する場合，画像内の顔の位置や大きさは未知であるため，位置やスケールを変えて切り出したサブウィンドウを識別器に通して，顔か否かの判別を行う必要がある．図 2 のように，識別器のサイズは決まっているため，入力画像を縮小しながら画像内のあらゆる位置やスケールに対応した探索（ピラミッドスキャン）を行う必要がある．このようにピラミッドスキャンを行うと識別回数は数万回にもなり，ビデオレートでの顔検出は難しくなる．また，ステレオカメラを用いて顔検出を行った場合，異なる視点で撮影された複数枚の画像を探索することで検出精度の向上が期待できるが，処理コストが倍増してしまう．

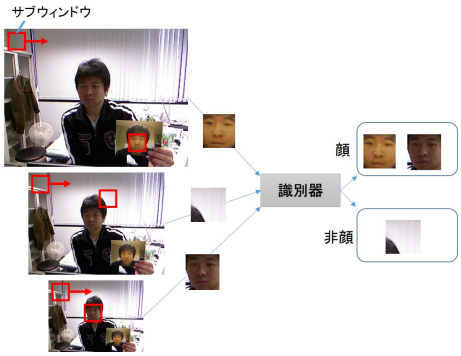


図 2 ピラミッドスキャン

識別回数と検出速度は比例関係にあり，識別回数を削減することができれば大幅な速度の改善が期待できる．提案手法では，ステレオカメラを用いて得られる奥行き情報を活用して探索範囲やスケールを限定し，識別回数の削減によって顔検出の高速化を実現する．ステレオカメラを利用することで，検出精度を向上させながら，単眼カメラよりも高速な顔検出を行うことができる．また，スケールを限定した探索によって，写真やカレンダーに写っている顔など実物の顔と大きさが異なる顔を検出しないといった効果も得られる．

2.1 ステレオカメラを用いた高速顔検出法

本手法ではステレオ処理によって得られた距離情報を用いることで，探索する領域とスケールを特定し，識別回数を減少させることで，ビデオレートでの顔検出を実現する．基本アイデアは，次の 3 点である．

- スパースサンプリングによるステレオ処理の高速化
- 識別回数の削減による検出の高速化
- ステレオ画像を用いた検出精度の向上

2.1.1 スパースサンプリングによるステレオ処理の高速化

入力画像の全画素に対してステレオ処理を行うと，多大な計算時間となり，ビデオレートでの処理は難しい．そこで，スパースにサンプリングした画素に対してのみステレオ処理を行う手法を提案する．

スパースサンプリング

ここでは，処理時間とサンプル数を考慮したサンプリングの方法について述べる．まず，使用している識別器のサイズが 20×20 ピクセルであるため，図 3 の様にサンプル点の間隔を 20 ピクセルとする．予備実験により，このままのサンプル点に対して奥行き情報を求めると約 20msec の処理時間を要し，検出処理を含めた全処理をビデオレートで行うことが困難である．そこで，図 3 のようにサンプルを赤と青の 2 つのグループに分ける．これら 2 種類のサンプル配置をフレームごとに切り替えてステレオ処理を行う．以上のようにすることで 2 フレームに 1 回は最小サイズの顔を検出できることを保証している．

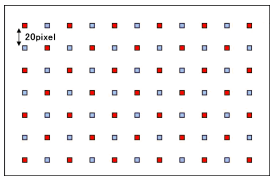


図 3 サンプル点の配置

重点サンプリング

画像上での顔が小さい場合，顔上のサンプル数が少なくなるので，距離が正しく計算でできなかった場合に検出できないことがある．一方，画像上での顔が大きい場合必要以上のサンプル数となる．このような問題を軽減するために重点サンプリングを行う．重点サンプリングは，前フレームで顔が検出された場合に行い，図 4 のようなサンプル配置とする．前フレームで図の矩形のように顔が検出された場合，その付近にサンプル点を 16 点配置する．こうすることで，小さい顔が検出されたところではサンプル間隔が細かく，大きい顔が検出されたところではサンプル間隔が粗くなる．

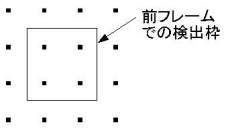


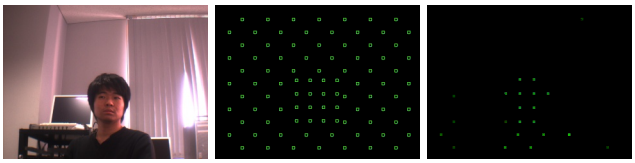
図 4 重点サンプリング

以上のサンプリング法によりステレオ処理を行った結果例を図 5 に示す．図 5(c) のサンプリングされた距離画像を元に探索領域とスケールを決定する．

2.1.2 識別回数の削減による検出の高速化

奥行き情報に基づいて，探索する領域とスケールを限定することで識別回数を削減する．処理の流れは次のようになっている．

(1) 距離値から画像上での顔サイズを推定する



(a) 入力画像　(b) サンプル配置　(c) ステレオ処理結果

図 5　サンプリングによるステレオ処理の結果例

(2) 顔サイズから探索領域とスケールを決定する
(3) 探索領域の統合を行う

画像上における顔サイズの推定

顔の大きさには個人差があるが，それほど大きな違いがない．そのため，ワールド座標系における実際の顔の大きさをあらかじめ決めておくことで，カメラから顔までの距離が分かれば画像上での顔の大きさを求めることが出来る．

図 6 に示すように，実際の顔の大きさを W_{size} とすると，画像上の顔の大きさ I_{size} は次式 (1) により推定できる．

$$I_{size} = \frac{f}{Z} W_{size} \qquad (1)$$

ただし，f はピクセル単位の焦点距離，Z はワールド座標におけるカメラから物体までの距離である．

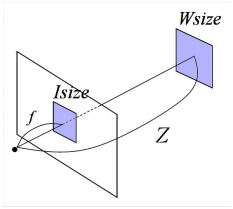


図 6　画像上での顔サイズの推定

図 7 は入力画像と距離画像の例である．得られた距離値から式 (1) により算出した画像上の顔サイズ（図中の青色の矩形）のみの探索を行う．

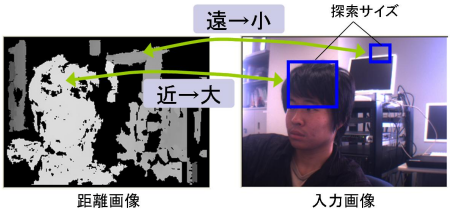


図 7　距離に応じた探索スケールの決定

探索領域とスケールの決定

図 5(c) の各サンプル点において，式 (1) により求められた顔サイズを用いて探索領域とスケールを決定する．探索領域は，図 8 に示すようにサンプル点から上下左右に顔サイズ分だけ離れた矩形領域とする．得られたサンプル点が顔のどこに位置しているかわからないため，このような探索領域とすることで，サンプル点が顔上のどこに位置していても，探索領域内に顔が含まれるようになる．探索するスケールは，識別器に通す際のサブウィンドウの大きさが式 (1) で求めた顔サイズとなるように決定する．

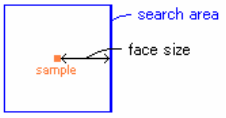
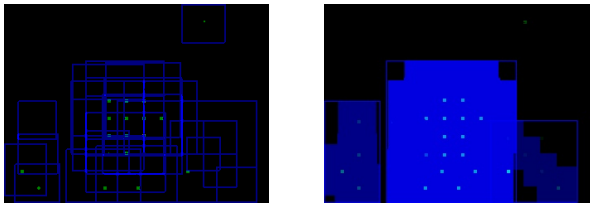


図 8　各サンプル点における探索領域

探索領域の統合

図 5(c) の距離画像における全てのサンプル点について，前述のようにして探索領域を決めると図 9(a) のようになる．画像中にある物体付近では，距離値のほとんど変わらないサンプル点が集まっているため，重なっている領域を距離値が近いものに限り一つの探索領域とする．また，この領域内を探索するスケールは，統合された領域の平均とする．

図 9(a) に対して，統合を行った最終的な探索領域は図 9(b) のようになる．図中のそれぞれの矩形領域内を求められた特定のスケールのみで探索する．また，ノイズによる無駄な探索領域を増やさないため，統合されなかった領域を排除する．顔が識別できる最小サイズのような場合であっても，顔上で得られたサンプルと体上で得られたサンプルが統合されて 1 つの探索領域となる．このように，顔や物体がある付近では，ほとんどの場合複数のサンプルが統合される．つまり，どのサンプルとも統合されないような独立したサンプルはノイズである可能性が高いためこれを除去する．



(a) 各サンプル点の探索領域　(b) 統合結果

図 9　探索領域の統合

2.1.3　ステレオ画像を用いた検出精度の向上

本研究で顔検出時に用いている識別器は，正面顔画像のみによる学習で構築されている．そのため，ある程度横を向いた画像などでは顔と識別することができない．本研究で用いるステレオカメラでは，視点の異なる左右 2 枚の画像が得られる．このような 2 枚の画像での探索を行うと，一方の画像では顔が検出できない場合でも，もう一方の画像では検出できるケースがある．そのため，ステレオ画像を探索することによって検出率の向上が期待できる．また，提案手法では識別回数を大幅に削減することができるため，識別回数の減少に伴って必然的に誤検出の数も減少する．以上のことより，単眼カメラによるピラミッドスキャンに比べ提案手法では検出精度が向上すると考えられる．

2.2 実験

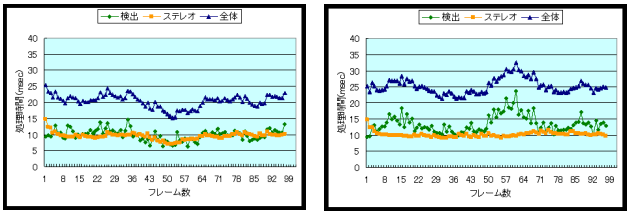
100 フレームほどの実画像系列（ステレオ画像 320 × 240 画素）を数種類用意し，全探索（ピラミッドスキャン）と提案手法による比較実験を行った．全探索では，識別できる最小顔サイズを 20 × 20 とし，倍率を 1.25 倍としている．一般的に，物体検出ではスケールを細かく見るほど検出率は上がるが，使用ライブラリのデフォルト値を用いている．提案手法では次のようにしている．式 (1) のワールド座標系における顔の大きさ W_{size} を 14cm と設定する．また，個人による顔の大きさの違いや距離計算時の誤差などを考慮し，各探索領域で 2 スケール分を探索するようにする．実験環境については，Pentium(R) 4 CPU 3.0GHz (2 CPUs)，1024MB RAM の PC を使用し，ステレオカメラとして Point Grey Research 社の Bumblebee を用い，顔識別器は OpenCV[12] のライブラリ関数を利用している．

表 1　探索サイズ

	全探索	提案手法
開始サイズ	20 × 20	式 (1) の I_{size}
スケール倍率	$1.25^N (N = 0, 1, 2, ...)$	$1.25^N (N = 0, 1)$

2.2.1 処理速度

用意した 2 種類の画像系列に対し，提案手法による顔検出を行った．系列 1 は 1 人が映っているもので，系列 2 は 2 人が映っているものである．図 10 に処理時間のグラフをそれぞれ示す．横軸はフレーム番号で，縦軸は処理時間 (msec) である．緑色 (◆) のグラフは検出にかかる時間，オレンジ色 (■) のグラフはステレオ処理にかかる時間であり，青色 (▲) のグラフは 1 フレーム全体の処理時間となっている．



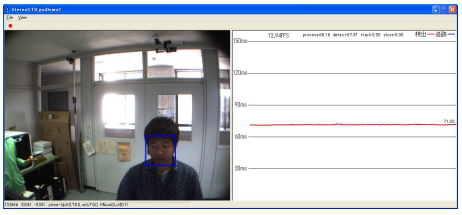
(a) 系列 1（1 人）　(b) 系列 2（2 人）

図 10　処理速度

図 10 より，ステレオ処理にかかる時間はほぼ一定で 10msec 程度となっており，検出にかかる時間によって全体の処理時間が変化しているが，ほとんどのフレームにおいて 33msec 以内で全ての処理が終わっている．系列 2 では 2 人の人が映っているため，系列 1 に比べて探索領域が増え，検出に要する時間が全体的に多くなっている．

国際会議や国内大会において，様々な環境で提案手法を用いた顔検出のデモを行ったが，提案手法での 1 フレームに要する平均処理時間はおよそ 25msec となっている．また，片方のカメラ画像に対してのみ全探索を行った場合，識別回数に変化はないため処理時間はほぼ一定で，1 フレームに要する処理時間は約 70msec であり，左右カメラ画像に対して単純に全探索を行うと処理時間は約 140msec となる．デモ時は，図 11 のように検出結果と処理時間のグラフをリアルタイムで表示している．処理時間のグラフでは，縦軸が処理時間をミリ秒で表し，横軸が処理時間の履歴を表しており，右端が現フレームの処理時間である．

実験やデモの結果より，2 枚のステレオ画像では，提案手法によって全探索に比べ 5 倍以上の高速化が行える．単眼カメラからの全探索と比べた場合，提案手法ではステレオ画像を探索しながら 3 倍近く高速化された顔検出が行え，ビデオレートでの処理が可能となった．



(a) 片側カメラからの全探索

(b) 提案手法

図 11　デモアプリケーション画面

2.2.2 検出精度の実験結果

用意した 3 種類の画像系列に対して，

- 単眼画像からの全探索（右画像を使用）（全探索）
- 提案手法で右画像のみの探索（提案手法 1）
- 提案手法で左右の画像を用いた探索（提案手法 2）

の計 3 通りでの検出実験を行った．系列 3 は一人の人物が移動しているようなもので，系列 4 は映っているのは 1 人で動きが早いもの，系列 5 は 2 人の人物が映っているようなものを用いた．提案手法 1 の場合は，各探索領域において式 (1) で求めた I_{size} とその 1.25 倍の計 2 スケールを探索，提案手法 2 の場合は，右画像で I_{size}，左画像でその 1.25 倍の計 2 スケールを探索した．比較実験の結果を表 2 に示す．表の数値は 101 フレーム中，顔を検出できたフレーム数を表している．() 内の数値は 101 フレーム中にあった誤検出の合計数を表している．系列 5 では 2 人の人物が映っているため，人物 a，人物 b に分けて検出数を数えた．

表 2 より，全ての場合に単眼画像からの全探索に比べると提案手法の検出数が多くなっていることが分かる．単眼画像からの全探索では検出できているのに提案手法で検出

表 2　手法による検出数の比較

画像系列	3	4	5a	5b
全探索	82(8)	52(20)	54(22)	83(22)
提案手法 1	87	77	55	92
提案手法 2	87(2)	87(7)	55(2)	92(2)

できなかった例としては，ステレオ画像から顔までの距離を正確に求められず（例えば，画像の右端の方は視差の関係で正確な距離が出せない）顔を含むような探索領域が正確に設定されていない場合などがあった．また，誤検出については，識別回数の削減にともない提案手法の方が大幅に減っている．提案手法 2 は，全探索に比べ検出数が平均して約 20%向上しており，全探索の誤検出を約 80%削減できている．

全探索，提案手法 1 を比べると，同じ右画像のみによる探索であるが，提案手法の検出数が多くなっている．単眼画像からの全探索の場合，探索サイズを 1.25 倍ずつ大きくしているため，顔の大きさが 2 つのスケールの間であるような場合に検出されにくいといったことが起こる．一方，提案手法では距離に応じたサイズの探索をしているため，そのときの顔の大きさにあったスケールでの探索となり，全探索のような問題は起こらない．そのため，片方の画像だけでの探索でも，的確なスケールを探索しているため，提案手法では全探索よりも検出率が向上している．

提案手法 1，2 の検出数を比べると，系列 3，5 では違いが見られなかったが，系列 4 については検出数が多くなっている．系列 3，5 では映っている人がほぼ正面を向いていて，動きも早くなく，比較的検出されやすい状況であったため，差が見られなかったと考えられる．一方，系列 4 では検出数が 1 割以上増加しており，右画像で検出できなかった顔が左画像で検出できている．これは，左右カメラの特性や視線の違いによるものだと考えられる．このように，一方の画像で検出できなかった顔がもう一方の画像では検出できるケースがあるため，ステレオ画像を用いた探索が有効であるといえる．

2.2.3　その他の検出結果

図 12 は，3 人の人物が写っている動画シーケンスに対して，提案手法による顔検出を行った結果例である．図では，左側に右カメラからの入力画像と検出結果，右側に入力画像に対する距離画像と探索領域を示している．図のように 3 人の人物が写っているような場合では，それら人物の部分にそれぞれ探索領域が設定されていることがわかる．

図 13 は，実物と顔写真が写っている動画シーケンス（100 フレーム）に対し，全探索と奥行き情報を利用して探索を行う提案手法を用いて検出を行った比較結果である．全探索では，顔の大きさにかかわらず検出されているのに対し，提案手法では，実物の顔のみが検出されていることがわかる．顔写真の位置で算出された顔サイズと顔写真のサイズが大きく異なっているため，顔写真は検出されない．提案手法では，全フレームにおいて写真の顔が検出されることはなかった．このように，奥行きに応じたスケールのみの探索を行う本手法を用いることで，顔写真など実際の顔の大きさと異る顔を検出せずに実物の人のみを検出することができる．

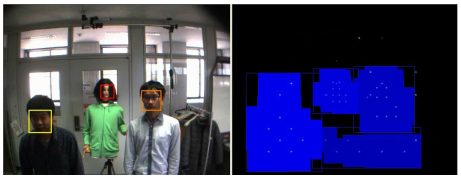
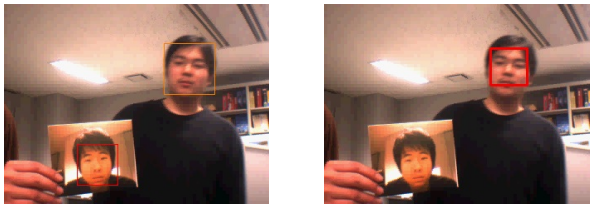


図 12　複数人物の検出例（左：入力画像と検出結果，右：距離画像と探索領域）



(a) 全探索　　(b) 提案手法

図 13　実物と顔写真の検出例

3. 顔検出と色弁別度追跡法を組み合せた顔追跡システム

本章では，2 章にて提案した高速顔検出法と色弁別度追跡法を統合することで，自動的に追跡が開始できる高速かつ安定な顔の検出・追跡システムの構築方法について説明する．顔検出を毎フレーム行うことで顔領域を取得することは可能であるが，正面顔画像のみで構築された識別器では顔がカメラに正対していなければ検出することができず，継続的に顔領域を取得することができない．そこで本研究では，識別器による顔検出と色情報を用いた追跡を統合する．また，提案システムでは，奥行き情報から画像上での顔の大きさを推定することで追跡を安定化させ，顔検出で取得した領域からターゲット（顔）と背景（非顔）の色ヒストグラムを構築し，背景には多く含まれず追跡しやすいターゲット色を選びながら追跡モデルを自動的にアップデートすることで，照明変動や周辺環境の変化に頑健な追跡を実現する．

3.1　色弁別度追跡法

色弁別度追跡法 [13], [14] は，色に基づく弁別性マップを構築し，弁別度の高い領域に楕円フィッティングすることで追跡を行う手法である．画素毎に弁別度を算出するため，ワイヤオブジェクトの追跡やオクルージョンに対する頑健性を持っている．また，テーブル化することで高速化

した弁別度算出によって弁別性マップを構築しているため，非常に高速な追跡処理が行える．また，弁別度は最近傍識別によるターゲット検出としても利用でき，検出処理と追跡処理を統合することでターゲットの検出・追跡システムを構築している．

対象色らしさを表す量を「弁別度」，全画素の弁別度を画像にしたものを「弁別性マップ」と呼ぶ．色弁別度は対象・非対象のプロトタイプ集合と入力との距離をそれぞれ求め，これら 2 つの距離から計算される量であり，以下の性質を持つ．

- (0,1) の範囲の実数値である．
- 0.5 がちょうど最近傍識別での識別境界となる．
- 対象プロトタイプ上で最大値 1 となる．
- 非対象上で最小値 0 になる．

弁別度 d は次式のように計算できる．

$$d = \frac{D_{non-target}}{D_{target} + D_{non-target}} \qquad (2)$$

但し，D_{target} は色空間内で対象色として与えられた最近傍ターゲットプロトタイプと入力画素値との距離，$D_{non-target}$ は非対象色として与えられた最近傍非ターゲットプロトタイプと入力画素値との距離を表す（図 14）．弁別度は，ターゲットとの類似性と非ターゲットとの相違性をベイズ則によって統合することで表現され [15]，対象追跡を行う際の評価尺度として用いることができるだけでなく，最近傍識別器による色ターゲット検出という優れた対象検出も同時に行える画像特徴となっている．

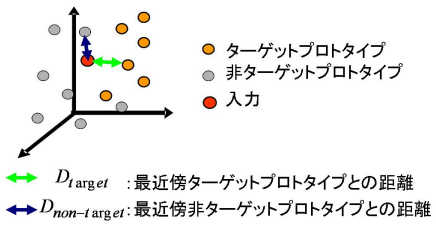
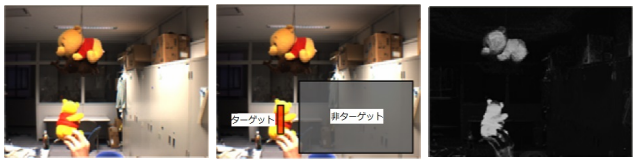


図 14 色空間内での距離表現

図 15 に弁別性マップを構築した例を示す．図 15(b) は，入力画像に対して手動で教示を行っている様子を示しており，赤色の矩形領域をターゲット，灰色の矩形領域を非ターゲットとしてそれぞれ指定している．図 15(c) では，弁別度を 0〜255 の値に拡大して表示している．ターゲットとして教示したぬいぐるみの領域が顕著に現れていることがわかる．

また，弁別性マップを構築する際，入力画像の全ての画素において式 (2) を適用して弁別度を算出すると，処理に要する時間が多くなってしまうため，量子化された色空間すべての弁別度を算出した弁別度 LUT（Look Up Table）を予め構築することで，弁別度算出の高速化を行っている．



(a) 入力画像　(b) 教示範囲　(c) 弁別性マップ

図 15 弁別性マップの構築 [14]

3.2 適応的カラーモデルを用いた顔検出・追跡システム

先行研究 [14] における色ターゲット検出・追跡システムの処理手順を図 16 に示す．

先行研究では，顔の追跡を行う場合，あらかじめ肌色情報（ターゲットプロトタイプ）を教示し，ターゲット色に基づく検出を行ってるため，教示した色と似た色が画像中に入ってくると誤ってターゲットとして検出し，追跡されてしまう問題がある．提案手法では，肌色情報ではなく 2 章にて提案した高速な顔検出法を用い，入力画像中から顔だけを正確に検出することでこの問題を解決する．

また先行研究では，照明環境の変化や人種，個人差による肌色の違いなどによって，あらかじめ教示したターゲット色と異なる追跡対象者を検出できなかったり，検出に成功したとしても追跡が安定に行えない問題がある．提案手法では，検出された顔領域に基いてプロトタイプの登録を行うことで，環境に合わせたカラーモデル自動的に構築する．

さらに，追跡中に背景に対象と似ている色が隣接してしまうと，追跡対象周辺の画素も高い弁別度を持ち，追跡楕円が広がりすぎてしまうという問題がある．提案手法では，顔追跡に特化しているため，ステレオ視によって得られた奥行き情報から画像上での顔の大きさを推定し，この顔サイズより追跡楕円が大きくならないように制限することで，安定な追跡が行える．また，先行研究では弁別度 LUT の更新を行っていないため，照明変動や背景の変化に適応できないのに対し，提案手法では，顔検出が成功した際に LUT の更新を行うことでこの問題を解決する．

提案手法の処理の流れを図 17 に示す．また，提案手法の要点は次のようになっている．

- 識別器を用いた顔検出と弁別度追跡を組み合わせ，顔だけを正確に追跡する
- 顔検出で得られる顔領域に基づいて構築された色ヒストグラムによって，適応的な弁別度 LUT の構築と更新を行う
- 顔までの距離に基いて画像上での大きさを制限し，追跡を安定化する

3.2.1 プロトタイプの自動登録による適応的カラーモデルの構築

提案システムでは，顔識別器により正確な顔領域が得られるため，顔領域を基にプロトタイプの教示を行うことで，その時の環境に合わせたカラーモデルの LUT を自動的に

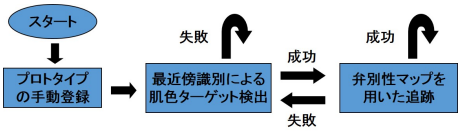


図 16　色ターゲット検出・追跡システムの流れ

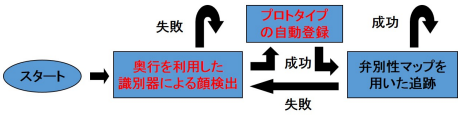


図 17　提案手法における顔検出・追跡の流れ

構築することができる．

プロトタイプの登録から弁別度 LUT 構築までの処理の流れは図 18 のようになっている．まず，検出枠の矩形領域内（赤色領域）をターゲット（顔）領域，それ以外（青色領域）を背景領域とし，各領域ごとに，領域内の画素値を色空間に投票して，ある色がいくつあるのかを表す度数ヒストグラムを作る．図 18 右上のヒストグラムの横軸が色空間，縦軸が各色の度数を表している．次に，構築されたヒストグラムにおいて，ターゲットと非ターゲットの度数を比べ，度数が大きい方をそのプロトタイプとして距離テーブルに登録する．つまり，非ターゲット領域には少なくターゲット領域に多い色がターゲットプロトタイプとなり，残りの色が非ターゲットプロトタイプとなる．図 18 の右中，ヒストグラムの下の赤色矢印がターゲットプロトタイプ，青色矢印が非ターゲットプロトタイプを表している．最後に，2 つの距離テーブルから弁別度 LUT を構築する．このような学習を行うことで，できるだけ多くの背景色を教示しつつ，背景には少ないターゲット色を学習することができるため，安定した追跡が行える．

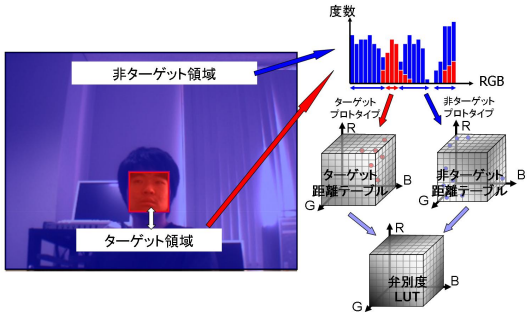


図 18　プロトタイプの自動学習方法

3.2.2　奥行き情報と適応型カラーモデルを用いた追跡の安定化

追跡楕円の制約

先行研究 [14] でも示されているように，追跡時，背景に対象と似ている色が現れ対象と隣接してしまうと，追跡が失敗してしまうという問題がある．先行研究では，ステレオ追跡によって対象までの距離を測ることができる利点を活かし，対象までの距離と楕円サイズを対応付けることでこの問題を解決できることが示唆されていが，追跡が安定している状態でこの対応付けを行うフレームを手動で決定する必要がある．提案手法では，ステレオ視によって得られた距離情報から画像上での顔の大きさを推定し，楕円サイズが画像内の顔サイズと同程度となるように制限することでこの問題を解決する．

弁別度 LUT の更新

追跡中にターゲットと似た色の未学習の背景が入ってきた場合に，追跡が不安定になることがある．本研究では，追跡中に弁別度 LUT を更新することによってこの問題を解決する．弁別度 LUT の更新を行うには，正確なターゲットの領域を知る必要がある．例えば追跡が失敗し，追跡楕円が対象から外れてしまったときに楕円内をターゲットプロトタイプとして登録してしまうと，誤った弁別度 LUT が構築されてしまう．そこで提案手法では，追跡時にも顔検出を行い，顔が検出された時のみ更新処理を行う．追跡中の顔検出は前フレームの対象（顔）の周辺領域のみで行い，また前述したように対象までの距離から顔サイズが求められるため，そのスケールのみの探索を行う．対象までの距離は左右カメラの追跡楕円の中心のずれから計算でき，追跡対象の周辺領域だけを 1 つのスケールのみで探索を行うため，ほとんど計算コストのかからない高速な顔検出ができる．検出が成功した場合，3.2.1 項で説明した処理と同様にプロトタイプの登録を行い，弁別度 LUT を再構築する．このとき，再構築の計算コストを抑えるため，非ターゲット領域はターゲット領域の 2 倍程度としている．

3.2.3　提案システムの構成

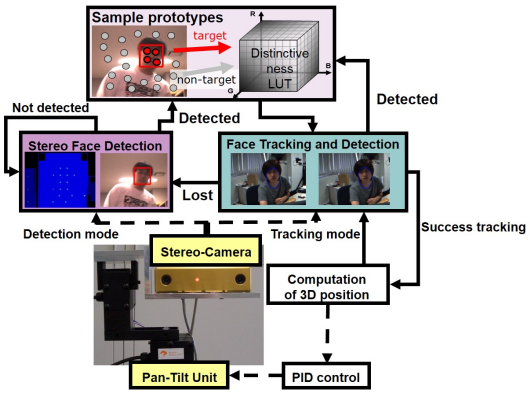


図 19　提案システムの構成

提案手法を用いて，能動ステレオカメラを用いた顔追跡システムを構築した．ステレオカメラが図 19 に示すように PTU 上に搭載されており，追跡対象を画像中心に捉えるよう PID 制御により自動的に追尾撮影を行う．PID 制御は「鮮明な画像撮影のための高速追従カメラシステム [16]」で用いられている方式を採用している．使用機材は，計算機が Intel Core i5-750 の CPU を搭載し

た PC，ステレオカメラが PointGrey Research 社の Bumblebee2，Pan-Tilt Unit(PTU) が Directed Perception 社の model PTU-46-17.5 である．高速に弁別度を参照するために，弁別度 LUT の色空間を $64 \times 64 \times 64$ に量子化している．

提案システムでは，まず顔検出を行い，顔が検出されると弁別度 LUT の構築と追跡初期位置が設定され，追跡へと移行する．また，追跡中にも顔検出が行われ，検出成功時に弁別度 LUT の更新が行われる．追跡が失敗すると再び顔検出へと切り替わる．追跡楕円内の弁別度が急激に下がった場合に追跡失敗と判断している．また，識別器を用いた顔検出によって正面顔のみを検出するため，本システムに興味のある人を検出・追跡し，興味の無い人，つまり，システムの前を素通りする人は検出・追跡しないというインタラクションを実現している．

3.3 実験

3.3.1 プロトタイプ登録方法の比較

照明環境や追跡対象の異なる 4 つの画像系列を用意し，プロトタイプの教示を手動（弁別度 LUT が固定）で行った場合と各系列で検出された顔領域から自動（弁別度 LUT が系列ごとに変動）で行った場合の検出・追跡実験を行った．系列 1〜3 は照明環境が異なり，系列 4 は追跡対象が異なっている．また，手動登録では系列 1 と同様の環境で色の教示を行っており，顔検出の結果は追跡楕円の初期位置の設定にだけ用いられる．

実験結果の一部を図 20 に示す．上段から順に系列 1〜4 の結果を示しており，各段において，左図は顔が検出されたフレームをカラー画像で示しており，中図は弁別度 LUT 固定，右図は弁別度 LUT 変動時の追跡の様子を弁別性マップで示している．安定した追跡を行うためには，顔領域の弁別度が高く，背景領域の弁別度が低いほど理想である．また，青色の楕円が対象領域を表す追跡楕円である．

系列 1 と同様の環境で手動登録された弁別度 LUT 固定の場合では，環境の異なる系列 2〜4 の時，顔領域の弁別度が小さくなっていたり背景領域の弁別度が高くなってしまっていることが分かる．一方，各画像系列において顔検出の結果から弁別度 LUT が適応的に変動する場合では，すべての画像系列において背景領域の弁別度が低く顔領域の弁別度が高くなっていることが分かる．表 3 に，各系列の追跡開始後 100 フレーム間において，追跡楕円内の弁別度の平均値を比較したものを示す．表から，系列 1 では両方とも弁別度が高くなっているが，系列 2〜4 では弁別度 LUT 固定の場合に比べ変動するほうがより高い弁別度となっていることが分かる．

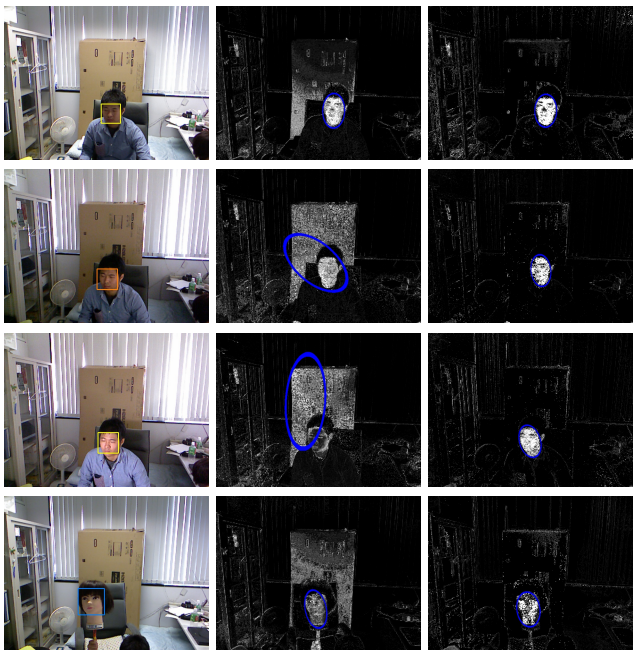
表 4 に，各系列の追跡開始後 100 フレーム間における楕円パラメータ（長軸，短軸）の標準偏差を示す．用いた画像系列では，ターゲットに奥行きの変化がほとんどないものとなっているため，ターゲットの大きさを表す楕円の長軸と短軸が一定であるほど安定した追跡であるといえる．弁別度 LUT 固定の場合，系列 1，4 では楕円パラメータの誤差が小さく安定した追跡が行えているが，系列 2，3 では楕円パラメータの誤差が大きくなっており安定した追跡が行えていないことが分かる．系列 2 については，顔領域の弁別度はある程度高いものの背景の弁別度も高くなってしまっているため追跡が不安定になっている．系列 3 については，顔領域の弁別度が小さくなっており追跡が失敗してしまっている．系列 4 については，顔領域と背景の弁別度が同じような値になっているが，弁別度が小さい髪領域によって分断されているため追跡楕円が広がらずに追跡が行えていた．一方，弁別度 LUT 変動の場合では，すべての系列において楕円パラメータの誤差が数ピクセルとなっており，安定した追跡が行えていることがわかる．

表 3 追跡楕円内の弁別度の平均値

	系列 1	系列 2	系列 3	系列 4
弁別度 LUT 固定	0.61	0.45	0.30	0.33
弁別度 LUT 変動	0.66	0.60	0.60	0.48

表 4 LUT 固定・変動時の楕円パラメータの標準偏差（単位：画素）

		系列 1	系列 2	系列 3	系列 4
弁別度 LUT 固定	長軸	5.21	43.86	34.01	6.19
	短軸	1.07	29.43	31.87	1.97
弁別度 LUT 変動	長軸	4.42	4.12	5.82	6.22
	短軸	1.61	1.29	1.49	4.41



検出フレーム　弁別度 LUT 固定　弁別度 LUT 変動

図 20 手動登録と自動登録の比較

表 5　安定化処理時の追跡楕円パラメータの標準偏差（単位：画素）

	x 座標	y 座標	長軸	短軸
安定化処理なし	93.61	20.10	49.38	28.98
楕円制約	2.13	1.13	3.70	0.49
弁別度 LUT 更新	1.38	1.24	1.34	0.60

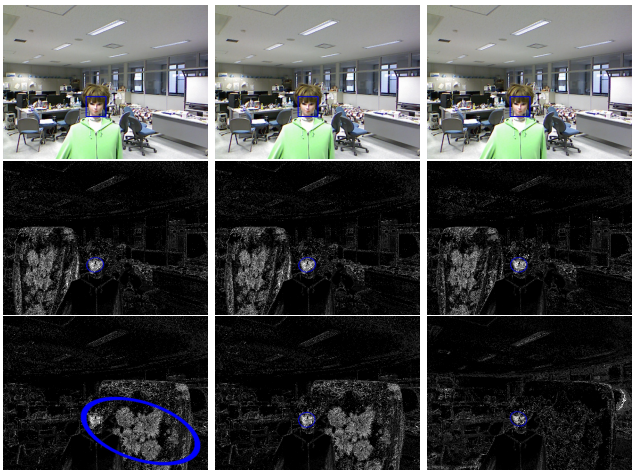
3.3.2　追跡中の安定化処理の有無による比較

追跡対象が固定されており，背景領域が変化するような 300 フレームの画像系列を用い，追跡楕円サイズの制約，弁別度 LUT の更新について，処理の有無による比較を行った．この画像系列は，追跡開始時には存在しない顔と似た色の背景領域が混入してくるようなものとなっている．弁別度 LUT の更新は，顔検出が成功した際に随時更新されるが，顔検出を 30 フレーム間隔で行うようにしている．

図 21 は，(a) 安定化処理を行わなかった場合，(b) 楕円制約を行った場合，(c) 弁別度 LUT の更新を行った場合の追跡結果の一部をそれぞれ示している．上段は顔が検出されたフレームをカラー画像で示しており，このフレームにおいて顔領域から弁別度 LUT が自動構築される．2，3 段目は追跡の様子を弁別性マップで示している．

安定化処理を行わなかった場合では，顔と似た色の背景が顔領域に近づき，追跡楕円が広がってしまっている．一方，楕円制約を行った場合では，安定した追跡が行えている．また，弁別度 LUT の更新を行った場合では，新たに混入してきた背景下で弁別度 LUT の更新が行われるため，背景領域の弁別度を低く保つことができ，安定した追跡が可能となっている．

また，追跡中の楕円パラメータ（x 座標，y 座標，長軸，短軸）について，その標準偏差を表 5 に示す．対象が固定されているため，楕円パラメータの標準偏差が 0 に近いほど追跡が安定していることを意味するが，安定化処理を行った場合では，安定した追跡が行えている．



(a) 安定化処理なし　(b) 楕円制約　(c) 弁別度 LUT 更新

図 21　安定化処理による追跡結果の比較

表 6　CAM Shift と提案手法の追跡楕円パラメータの標準偏差（単位：画素）

	x 座標	y 座標	長軸	短軸
CAM Shift	96.53	27.30	58.36	89.56
提案手法	1.78	1.38	1.36	0.35

3.3.3　他の追跡手法との比較

前述した追跡の安定化処理を両方行った提案手法と CAM Shift 法 [12], [17] による顔の追跡を行った．CAM Shift 法は，色ヒストグラムを用いて追跡を行う Mean Shift 法を拡張し，対象の大きさや姿勢の変化に対応した追跡手法であり，対象領域を楕円として得ることができる．実験では，顔検出で得られた矩形領域内で色ヒストグラムを構築することで，提案手法と同様に自動的に追跡が開始できるようにした．また，画像系列は前項と同じものを用いた．

図 22 に追跡結果の一部を示す．CAM Shift では，対象を表す楕円と中心位置を示している．CAM Shift では，顔検出の結果から対象の情報だけの色ヒストグラムを構築しているため，髪の色が混入して頭部全体が対象領域として追跡されている．また，対象と似た色の背景が重なると追跡楕円が広がってしまい，対象領域を正しく追跡できていない．一方，提案手法では安定した追跡を行えている．追跡開始から 300 フレーム目までの楕円パラメータ（x 座標，y 座標，長軸，短軸）の標準偏差を表 6 に示す．

(a)CAM Shift　(b) 提案手法

図 22　CAM Shift 法と提案手法の比較

4.　顔検出と 6D K-means Tracker を組み合わせた髪追跡システム

本章では，追跡手法として奥行き情報を用いた 6D K-means Tracker を提案し，2 章で提案した顔検出法と組み合わせた髪追跡システムの構築方法について述べる．本システムでは，近年広く普及している高速かつ高精度に奥行き情報を取得できる Kinect を用いる．

髪領域は，スタイルや姿勢変化による見えの変化が大きく，形状モデルを利用しにくいため画素単位の検出・追跡手法が望ましい．また，髪領域のテクスチャ情報は乏しく，

東洋人では髪の色が基本的に黒・灰・白であり色彩情報も乏しく，背景に同じ色が存在する可能性も高いため，色情報だけでなく距離情報を用いることで安定に髪領域と背景を分離できると考えられる．

前章で述べた色弁別度追跡法は，色情報のみでターゲットモデルや背景モデルを構築し，LUT を用いた高速化を行っているため，常に変化する奥行き情報をモデルに導入することは難しい．一方 K-means Tracker は，フレームごとにクラスタ中心の更新を行っているため，奥行き情報の導入が容易であり，モデルを次元拡張することで追跡の安定化が期待できる．

4.1 K-means Tracker

K-means Tracker[18], [19] は，画像上の追跡対象と背景の両方に対して複数のクラスタ中心を割り当て，3 次元の色空間と 2 次元の座標空間を合わせた 5 次元特徴空間における K-means クラスタリングによって，サーチエリア内の各画素をターゲットと非ターゲットに分けることにより追跡を行う手法である．追跡対象に割りあてられたクラスタ中心をターゲットクラスタ中心，背景に割り当てられたクラスタ中心を非ターゲットクラスタ中心と呼ぶ．特徴空間内で，これらクラスタ中心とサーチエリア内の各画素との距離を計算し，その距離に基づいてターゲットか非ターゲットのラベルが入力画素に付けられる．

K-means Tracker では，ターゲットクラスタ中心は画像中の追跡対象上に，非ターゲットクラスタ中心は，画像上で追跡対象を取り囲むよう背景に応じて複数配置（適応的な非ターゲットクラスタ中心の配置法 [18]）される．追跡対象が動けば，クラスタリングの結果，ターゲットクラスタ中心の重心位置が更新され，非ターゲットクラスタ中心が再配置される．これを毎フレーム繰り返すことで，動画像上での対象追跡を行っている．

また，咸ら [20] は特徴空間に 1 次元の奥行き情報を追加し，2 次元座標空間における距離をマハラノビス汎距離で評価する拡張 K-means Tracker を提案し，追跡の安定化を行っている．

4.2 6D K-means Tracker

K-means Tracker は画素単位でのクラスタリングを行い，可変楕円をターゲット画素にフィッティングさせることで，追跡対象の大きさや形状の変化にも追従することができるため，様々な形状の髪領域を追跡するのに有効である．

しかし，従来の K-means Tracker（以下，5D K-means Tracker とする）には次のような問題点がある．追跡対象と類似する色を持つ背景画素がサーチエリアに混入した場合，5 次元特徴空間ではその背景画素が誤ってターゲット画素としてクラスタリングされる可能性が高く，その影響によりターゲット領域やサーチエリアが不安定となり，追跡が失敗してしまう場合がある．また，空間的距離を画像上の画素単位で計算するため，ターゲットが小さいく写っている時は 2 次元座標における距離は小さくなり，ターゲットが大きく写っている時は 2 次元座標における距離は大きくなる．このように，画像内のターゲットサイズによって，5 次元特徴空間内における 2 次元位置特徴ベクトルの割合が変わることで，大きく写っているターゲットの画素がすべてターゲット画素としてクラスタリングされない場合や，背景画素が誤ってターゲットとしてクラスタリングされてしまう可能性がある．さらに，クラスタ中心の初期位置を手動で指定しなければならないという問題もある．

拡張 K-means Tracker では，5 次元特徴に 1 次元の奥行き情報をそのまま追加することで類似色背景問題に対応しているが，画像内ターゲットサイズ問題に対応するためマハラノビス汎距離を用いてクラスタリングを行うので，非ターゲットクラスタ中心を適応的に配置することができない．

そこで本論文では，各画素を実空間に変換した上で画素間の空間的近接性を評価する 6D K-means Tracker を提案する．これにより，5D K-means Tracker の問題を解決しつつ提案システムを構築する方法について述べる．提案システムの特徴は，以下のようになっている．

- 3 次元の色情報と 3 次元の実空間位置情報からなる 6 次元特徴空間における 6D K-means Tracker により，安定な追跡を行う．
- 顔検出と 6D K-means Tracker を組み合わせ，検出された顔領域に基づいて，ターゲット（髪・顔）と背景のクラスタ中心の初期特徴ベクトルを自動的に決定する．

4.2.1 6 次元特徴空間でのクラスタリング

本手法では，Kinect で獲得されたターゲット，背景までの奥行き情報とそれぞれの色情報を活用し，K-means Tracker を 6 次元に拡張する（図 23 参照）．

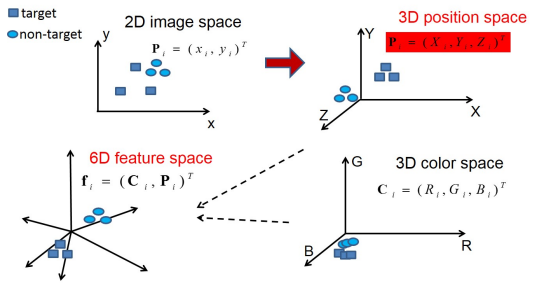


図 23 6 次元特徴空間

画素の位置 (x, y) を表現する 2 次元ベクトルを 3 次元カメラ座標空間 (X, Y, Z) に射影する．

$$\mathbf{p_2} = [\mathbf{x}\,\mathbf{y}]^{\mathrm{T}} \rightarrow \mathbf{p_3} = [\mathbf{X}\,\mathbf{Y}\,\mathbf{Z}]^{\mathrm{T}} \qquad (3)$$

また，追跡対象を構成する画素の色の類似性と空間的近接

性を同時に表現するために，各画素の特徴を 3 次元色空間 $\mathbf{c_3} = [\mathbf{R\,G\,B}]^{\mathrm{T}}$ と 3 次元カメラ座標空間 $\mathbf{p_3} = [\mathbf{X\,Y\,Z}]^{\mathrm{T}}$ からなる 6 次元ベクトル $\mathbf{f_6} = [\mathbf{c_3\,p_3}]^{\mathrm{T}}$ として扱う．2 つの画素 a, b 間の距離 $d(\mathbf{f_6^a}, \mathbf{f_6^b})$ は，色空間内と 3 次元カメラ座標空間内のユークリッド距離に重み α を用いて，次式のように定義される．

$$d(\mathbf{f_6^a}, \mathbf{f_6^b}) = ||\mathbf{c_3^a} - \mathbf{c_3^b}||^{\mathbf{2}} + \alpha||\mathbf{p_3^a} - \mathbf{p_3^b}||^{\mathbf{2}} \qquad (4)$$

5D K-means Tracker では，2 次元の画素位置で空間的距離を評価しているため，固定された α を用いると追跡が破綻する場合があり，画像上のターゲットサイズに応じて α を考慮する必要があったが，提案手法では，実空間での距離評価を行っているため，固定値の α で安定した追跡が可能となる．

各画素を 3 次元カメラ座標系に変換した上で空間的近接性を求めているため，ターゲットと類似した背景が空間的に離れていれば，それらを分離でき，ターゲットサイズが一定であれば，カメラまでの距離変化によって起こる画像内ターゲットサイズ問題も解決できる．

本手法では，クラスタが未知である注目画素 $\mathbf{f_6^u}$ からターゲットクラスタ中心 $\mathbf{f_6^T}$ への最短距離 D_T，非ターゲットクラスタ中心 $\mathbf{f_6^{NT}}$ への最短距離 D_{NT} は，以下のように定義される．

$$D_T = \min_{i=1\sim n}\{d(\mathbf{f_6^{Ti}}, \mathbf{f_6^u})\} \qquad (5)$$

$$D_{NT} = \min_{j=1\sim m}\{d(\mathbf{f_6^{NTj}}, \mathbf{f_6^u})\} \qquad (6)$$

ここで，n, m は，それぞれターゲットクラスタ中心と非ターゲットクラスタ中心の個数であり，n は追跡初期フレームで設定され（4.3 参照），m はフレーム毎に動的に設定される（4.2.2 参照）．D_T と D_{NT} を比較することによって，注目画素 $\mathbf{f_6^u}$ がターゲットクラスタか非ターゲットクラスタにクラスタリングされる．

$$\mathbf{f_6^u} \rightarrow \mathbf{Traget} \quad \mathbf{if} \quad \{\mathbf{D_T(t)} < \mathbf{D_{NT}(t)}\}. \qquad (7)$$

$$\mathbf{f_6^u} \rightarrow \mathbf{Non\text{-}Traget} \quad \mathbf{if} \quad \{\mathbf{D_T(t)} > \mathbf{D_{NT}(t)}\}. \qquad (8)$$

4.2.2 クラスタ中心の更新

本手法では，可変楕円によるターゲット領域の記述は従来と同様の方法で行われ，非ターゲットクラスタ中心の更新は，適応的な非ターゲットクラスタ中心の配置法 [18] が，拡張された 6 次元特徴空間において行われる．

ターゲットとしてクラスタリングされた画素集合はガウス確率密度関数で近似することができると仮定する．ターゲットとしてクラスタリングされた画素の分布のマハラノビス距離より可変楕円を求め，ターゲット画素を 95%含む時のパラメータ（長軸，短軸，傾き，重心）を用いてターゲットの形状が記述される．求めた楕円を一定倍数拡大した楕円が次フレームのサーチエリアとなる．

非ターゲットクラスタ中心は，サーチエリアの輪郭上を背景と仮定し，この輪郭上の画素を非ターゲットクラスタ中心の候補点として次のようにして配置される．

まず，輪郭上の画素の 1 点が非ターゲットクラスタ中心として登録される．次に，輪郭上の画素を順にクラスタリングし，ターゲットに近い画素が非ターゲットクラスタ中心として追加登録されていく．このように，輪郭上を走査してターゲットにクラスタリングされそうな領域の 1 点を非ターゲットクラスタ中心とすることで，各フレーム毎の背景に応じた非ターゲットクラスタ中心の個数や配置が決定される．

このように，サーチエリアとクラスタ中心を毎フレーム更新することで，画像内の追跡対象の大きさや形状の変化，背景の変化に対して安定な追跡が行える．また，適応的な非ターゲットクラスタ中心の配置により，特徴空間を 6 次元に拡張した提案手法では，奥行きによってターゲットと背景の分類が容易になるため，少ない非ターゲットクラスタ中心の数で背景をモデル化することが可能となり，次元拡張による距離計算のコスト増加を抑制することができる．

4.3 髪・顔・頭部領域の検出・追跡システム

図 24 に提案システムの構成を示す．初期クラスタ中心を顔検出によって得られた位置情報から取得するため，本システムでは，まず顔検出処理を開始し，顔が検出されれば追跡処理へ移行する．追跡が失敗した場合は顔検出処理へ移行する．提案システムでは，追跡中にターゲットとしてクラスタリングされた画素が極端に少なくなった場合，対象の追跡に失敗したと判断している．

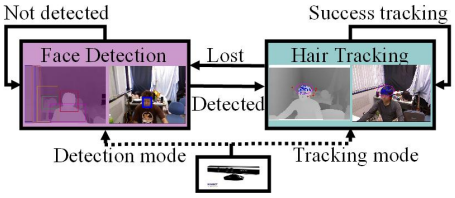
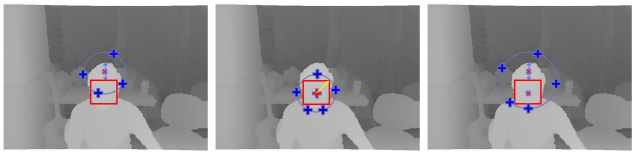


図 24 提案システムの構成図

顔検出に成功した場合，顔検出によって得られた矩形を用いて，追跡初期フレームのターゲット，非ターゲットクラスタ中心の 6 次元特徴ベクトルが対象領域の画素から自動的に設定される．髪を追跡する場合，図 25(a) に示すように，ターゲットクラスタ中心（赤点）は，顔として検出された赤矩形の上辺の中点から上方向のデプスのエッジ（デプスの値が極端に変わる）までの中点とする．非ターゲットクラスタ中心（青プラス）は，ターゲットクラスタ中心を円心，半径を顔サイズ（赤矩形の幅）とする青円上に自動的に配置（4.2.2 を参照）される．このように，ターゲットクラスタ中心は追跡対象から自動的に獲得されるため，髪の色に関係なく追跡を行うことができる．

提案手法は髪追跡システムを構築できるだけではなく，初期クラスタ中心の配置によって，顔追跡システムや，頭部（顔と髪の同時）追跡システムも簡単に構築することができる．顔を追跡する場合，図 25(b) に示すようにターゲットクラスタ中心は顔検出によって得られた赤矩形の中心，非ターゲットクラスタ中心は矩形の外接円上に配置される．頭部を追跡する場合は，図 25(c) に示すように 2 つのターゲットクラスタ中心を顔と髪の場合と同様に配置し，2 つのターゲットクラスタ中心の中点から赤矩形の右下までを半径とする円周上に非ターゲットクラスタ中心を配置する．頭部追跡を行った場合，頭部の回転運動などで顔領域をロストしても髪領域だけで追跡を継続することができ，顔領域が出現すると再び顔領域の追跡を再開することができる．



(a) 髪追跡　(b) 顔追跡　(c) 頭部追跡

図 25　追跡の初期化

4.4　実験

実験では，CPU が Intel Core i5-750，メモリが 2GB の汎用 PC を用いている．1 フレームの処理時間について，顔検出処理は約 14ms，追跡処理は，対象が一人の場合，約 10msec であった．また，式 (4) における α は，予備実験により決定した 0.4 に固定して全ての実験を行っている．4.2.2 節で述べたサーチエリアの倍率は 1.5 としている．なお，以下の実験結果の画像では，矩形は顔検出の結果，楕円はターゲット領域を表し，青色の画素はターゲットとしてクラスタリングされた画素，赤十字はサーチエリアの輪郭上に自動配置された非ターゲットクラスタ中心である．

4.4.1　類似色背景の場合

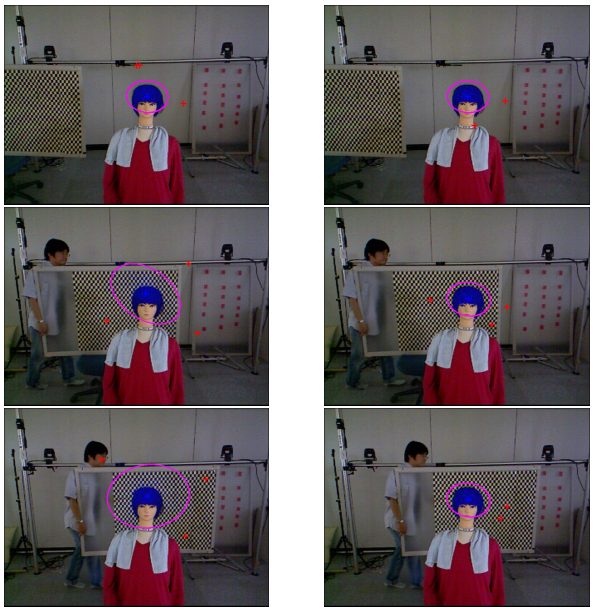
図 26 は，300 フレームの画像系列に対して 5D K-means Tracker と提案手法による髪の追跡を行い，安定性に関する比較を行ったものである．この画像系列では，追跡の安定性を比較するために，ターゲット（黒髪）を固定し，背景のみを変化（白と黒のチェッカーボードが横切る）させている．また，提案手法については，式 (4) の α を変化させて実験を行った．

図 26(a) は 5D K-means Tracker による追跡を行った結果であるが，背景にターゲットと類似色のチェッカーボードが近づいた時に，追跡楕円が広がってしまっている．一方，図 26(b) のように提案手法では，全フレームにおいて安定した追跡を行えていることが確認できた．

表 7 は，追跡中の楕円パラメータ（楕円中心の x 座標，y 座標，楕円の長軸，短軸）の標準偏差を比較したものである．ターゲットが固定されているため，楕円パラメータの標準偏差が 0 に近いほど楕円が一定で安定した追跡を行えていることを意味するが，提案手法は 1 ピクセル程度の誤差となっており，追跡結果が安定していたことがわかる．また，α を変化させた場合でも楕円パラメータの標準偏差は小さくなっており，追跡結果にほとんど差異はない．このように，提案手法は α の選択に影響されにくい結果が得られ，奥行き情報を導入することが，追跡性能の向上に貢献していることがわかる．なお，僅差ではあるがより最適であった $\alpha = 0.4$ を他の実験において用いている．

表 7　追跡楕円パラメータの標準偏差（単位：画素）

	x 座標	y 座標	長軸	短軸
5D K-means Tracker	4.33	4.6	13.19	12.06
提案手法 ($\alpha = 0.2$)	0.57	0.83	0.74	1.09
提案手法 ($\alpha = 0.4$)	0.57	0.77	0.78	1.03
提案手法 ($\alpha = 0.6$)	0.59	0.77	0.90	1.12
提案手法 ($\alpha = 0.8$)	0.67	0.74	0.96	1.12



(a)5D K-means Tracker　(b) 提案手法（$\alpha = 0.4$）

図 26　類似背景色の髪追跡結果例

4.4.2　ターゲットサイズが変わる場合

カメラから追跡対象までの距離の変化で，画像上のターゲットサイズが変わる場合の 5D K-means Tracker と提案手法による比較実験を行った．ターゲットサイズが徐々に大きくなるような 300 フレームの画像系列を用い，顔と髪からなる頭部領域の追跡を行った．この実験では，背景にターゲットと類似色の段ボールがある．

図 27 に追跡結果の一部を示す．従来の 5 次元特徴空間における追跡では，ターゲットサイズが小さい時には近く

にターゲットと類似色の背景があっても安定したクラスタリングが行える．しかし，ターゲットサイズが大きくなるにつれて 2 次元座標空間における距離が占める割合が大きくなるため，誤ったクラスタリングが起こりやすくなる．図 27(a) のように，一部段ボールの背景領域がターゲットの方が近いと判断され，誤ってターゲットとしてクラスタリングされてしまったことで追跡楕円が広がっている．このように，誤ってクラスタリングされた結果が以降のフレームに影響を与え，追跡が不安定になったり失敗してしまうことがある．一方，提案する 6 次元空間における追跡では，空間的近接性を 3 次元の実空間において評価しているため，ターゲットの大きさは常に一定の大きさとなる．図 27(b) に示すように，提案手法では終始安定した追跡を行うことができた．

また，追跡の安定性を評価するため，図 28(b) のように頭部領域を表す正解画像を用意し，各フレームにおいてターゲットとしてクラスタリングされた画素が頭部領域であった割合で評価した．

$$\text{ターゲット正解率} = \frac{\text{頭部領域にあるターゲット画素数}}{\text{ターゲット画素数}}$$

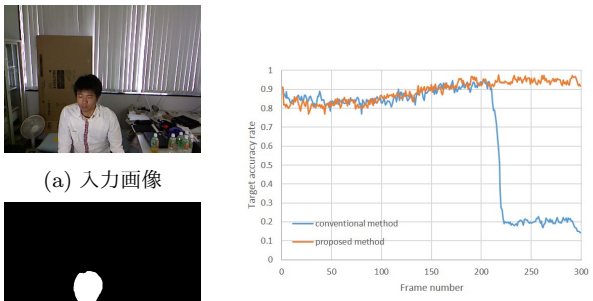
5D K-means Tracker と提案手法の各フレームにおけるターゲット正解率をグラフにしたものを図 28(c) に示す．5D K-means Tracker では，200 フレーム目付近から誤ったクラスタリングが増えているが，提案手法では，全フレームにおいて安定していることがわかる．

(a)5D K-means Tracker　　(b) 提案手法

図 27　ターゲットサイズが変化する場合の追跡結果

4.4.3　他の追跡手法との比較実験

提案手法と Particle Filter[12], [21] による髪の追跡を行った．Particle Filter は，確率密度分布を多数のサンプル（パーティクル）で近似し，確率の高い領域を対象とし



(a) 入力画像

(b) 正解画像

(c) ターゲット正解率

図 28　正解画像による精度評価

て追跡する手法である．実験では，各パーティクルに尤度を与える尤度関数を以下のように 2 種類定義した．

尤度関数 $L(d)$ は，ターゲットとパーティクルの距離 d に対して，平均を 0，分散を σ として持つような正規分布として定義する．

$$L(d) = \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{d^2}{2\sigma^2}) \tag{9}$$

ターゲットとパーティクルの距離について，色情報 $\boldsymbol{c} = [R, G, B]^{\mathrm{T}}$ のみを用いた d_1 と，それに奥行き情報 z を加えた d_2 として 2 種類定義した．d_1 は，ターゲットの色 $\boldsymbol{c}_t$ とパーティクルの色 $\boldsymbol{c}_p$ のユークリッド距離とする．

$$d_1 = ||\boldsymbol{c}_t - \boldsymbol{c}_p||^2 \tag{10}$$

d_2 は，ターゲットの奥行き z_t とパーティクルの奥行き z_p のユークリッド距離を加え，重み α を用いて次のように定義する．

$$d_2 = ||\boldsymbol{c}_t - \boldsymbol{c}_p||^2 + \alpha||z_t - z_p||^2 \tag{11}$$

実験では，σ を 30，α を 0.4，パーティクルの数を 3000 としている．また，ターゲットの情報は**??**で述べた髪領域のターゲットクラスタ中心の設定方法と同じように取得することで，提案手法と同様に自動で追跡が開始できるようにした．

実験に用いた画像系列は，対象が自由に動くようなものとなっている．追跡結果の一部を図 29 に示す．Particle Filter の結果では，青色の画素がパーティクルの位置を表し，黄色の十字がパーティクルの重心を表している．Particle Filter で色情報のみを用いた場合では背景の暗い部分と髪領域を区別することができず，追跡が失敗してしまっている．一方，奥行き情報を追加した場合は，髪領域の位置を正しく追跡できていることがわかる．

このように，奥行き情報を用いることで追跡手法の性能を大きく向上させることができる．追跡精度を評価するため，図 30(a) のように各フレームにおいて頭部の位置を表す基点（白色の十字）を両眉の中心に手動で設定し，各手法で得られた重心との距離（画像上でのユークリッド距離）を計測した．この距離が一定の値であるほど追跡が安定し

(a)PF 1（d_1）　(b)PF 2（d_2）　(c) 提案手法

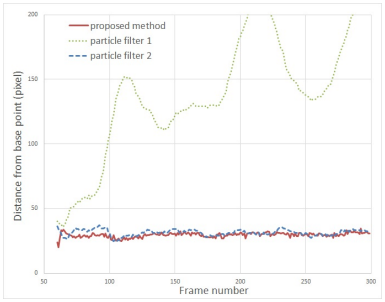
図 29　Particle Fileter と提案手法の比較

ているといえる．図 30(b) に，追跡開始から最終フレームまでの各手法で得られた重心と基点との距離をグラフにしたものを示す．また，各手法の標準偏差は表 8 のようになっている．奥行き情報を追加した Particle Filter 2 と提案手法は安定して髪領域の位置を取得できていることがわかる．

表 8　基点と重心の距離の標準偏差（単位：画素）

Particle Filter 1	Particle Filter 2	提案手法
46.60	2.37	1.84

(a) 頭部の基点（白色の十字）



(b) 基点と重心の距離（画素）

図 30　Particle Fileter と提案手法の精度評価

Particle Filter では，奥行き情報を追加することで，提案手法と同程度の精度で髪領域の位置を得ることができた．しかし，Particle Filter は，ランダムに配置されたパーティクルの位置でのターゲットらしさを確率として得ているため，髪の大まかな領域はわかるが，画素単位で髪領域を得ることはできない．一方，提案手法では，画素単位でターゲットか背景かの判別を行っているため，髪領域だけをより密に取得できる．また，Particle Filter は，尤度関数やパーティクルの数など多くのパラメータによって追跡性能が決まるため，安定した追跡を行うには処理コストも考慮した上でこれらを吟味する必要があるが，提案手法は，シンプルな処理で高速かつ安定した追跡を行うことができる．

4.4.4　様々な髪の追跡結果

図 31 は，髪の色やスタイルの異なる複数の対象を追跡した結果例である．3 体の対象（茶髪セミロング，金髪ショートのマネキンと黒髪ショートの人）に対し，検出・追跡を行なっている．このように，提案手法では髪の色やスタイルに関わらず追跡を行うことが可能であり，複数対象の追跡を行うこともできる．しかし，複数対象を追跡する場合，顔検出処理を常に行う必要があり，追跡中の対象が増えると追跡処理のコストも増えるため，対象が増えるに連れてビデオレートでの処理は困難となる．

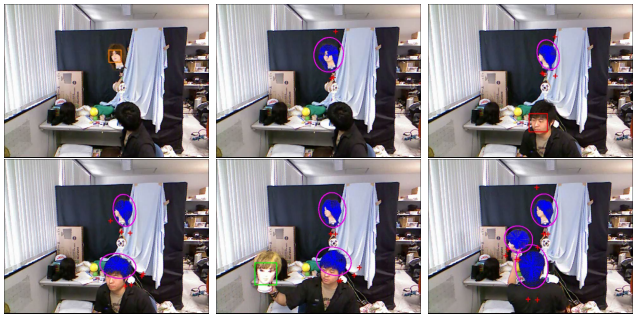


図 31　複数対象の追跡

4.5　頭部追跡

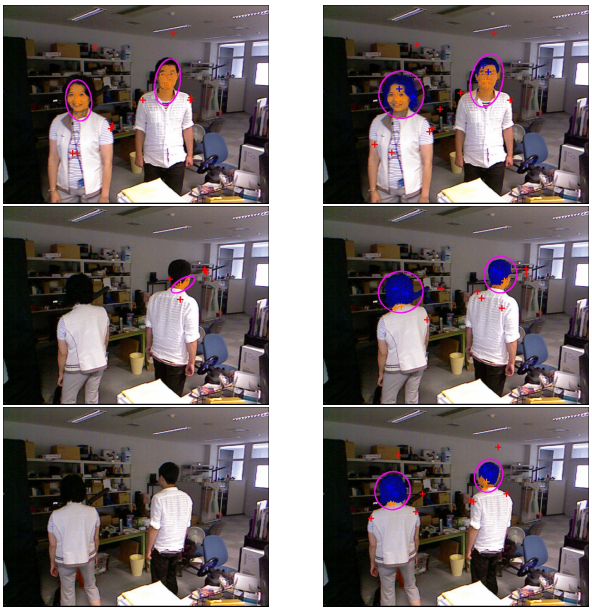
提案システムでは，顔領域と髪領域を同時に追跡することで頭部追跡が行える．図 32 は，提案手法を用いて顔だけを追跡した場合と，頭部（顔と髪）を追跡した場合の比較結果である．図 32(a) は顔領域だけの追跡を行っており，頭部の姿勢変化によって追跡が継続できなくなっている．一方，図 32(b) のように顔と髪を同時に追跡した場合は，失敗することなく追跡を継続することができている．このように，頭部の姿勢が変化してもどちらかの領域は必ずカメラに映るため，顔領域に加え髪領域を追跡することで，顔がカメラに対して後ろを向いた場合でも追跡を継続することができ，頭部の姿勢変化に関わらず頑健な追跡を行うことが可能となる．

図 33 は，髪追跡，頭部追跡の結果を 3D 表示したものである．このように提案手法では，画素単位の追跡により髪や顔領域の 3 次元形状をリアルタイムに得ることができる．

5.　おわりに

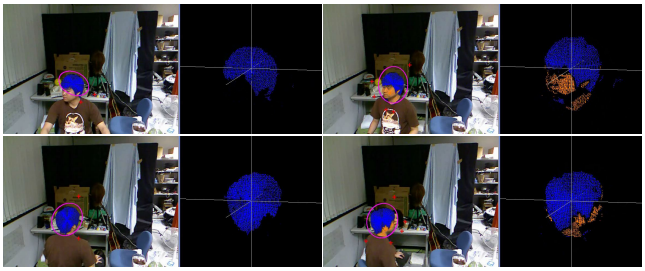
本論文では，奥行き情報を利用した顔検出の高速化と追跡の安定化について述べ，検出処理と追跡処理を統合することによって顔や髪領域の検出・追跡システムを構築した．

2 章にて述べた，奥行き情報を利用した識別回数の削減による検出の高速化手法は，既存の識別器を用いた検出を高速化させるものであり，単に顔検出だけでなく，一般的な物体の検出を高速化することができ，コンピュータビ



(a) 顔追跡　　(b) 頭部追跡（顔と髪）

図 32　顔追跡と頭部（顔と髪）追跡の比較



(a) 髪追跡　　(b) 頭部（顔と髪）追跡

図 33　髪，頭部（顔と髪）追跡結果の 3D 表示

ジョンやロボットビジョンの研究において役立つことが期待される．また，4 章にて提案した 6D K-menas Tracker は，適切な初期クラスタ中心を設定することで，髪領域だけでなく任意の非剛体物体をより安定に追跡することが可能であるため，コンピュータビジョンにおいて基礎的な処理である一般的な対象追跡に役立つことが期待される．

参考文献

[1] Viola, P. and Jones, M.: Rapid object detection using a boosted cascade of simple features, *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 1, pp. 511–518 (2001).

[2] 勢世，山下隆義，岡本卓也，川出雅人：高速全方向顔検出，画像の認識理解シンポジウム（MIRU），Vol. 2, pp. 271–276 (2004).

[3] Shaick, B.-Z. and Yaroslavsky, L.: Accelerating face detection by means of image segmentation, *Proceedings of EURASIP Conference focused on Video/Image Processing and Multimedia Communications*, Vol. 1, pp. 411–416 (2003).

[4] Cho, J., Mirzaei, S., Oberg, J. and Kastner, R.: Fpga-based face detection system using haar classifiers, *Proceedings of ACM/SIGDA International Symposium on Field Programmable Gate Arrays*, pp. 103–112 (2009).

[5] Hefenbrock, D., Oberg, J., Thanh, N., Kastner, R. and Baden, S. B.: Accelerating Viola-Jones Face Detection to FPGA-Level Using GPUs., *Proceedings of IEEE Symposium on Field-Programmable Custom Computing Machines*, pp. 11–18 (2010).

[6] Hg, R., Jasek, P., Rofidal, C., Nasrollahi, K., Moeslund, T. B. and Tranchet, G.: An RGB-D Database Using Microsoft's Kinect for Windows for Face Detection, *Proceedings of International Conference on Signal Image Technology and Internet Based Systems*, pp. 42–46 (2012).

[7] Huynh, T., Min, R. and Dugelay, J.-L.: An efficient LBP-based descriptor for facial depth images applied to gender recognition using RGB-D face data, *Computer Vision-ACCV 2012 Workshops*, Springer, pp. 133–145 (2013).

[8] Microsoft: Face Tracking SDK. `http://msdn.microsoft.com/en-us/library/jj130970.aspx`.

[9] Yacoob, Y. and Davis, L. S.: Detection and analysis of hair, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 28, No. 7, pp. 1164–1169 (2006).

[10] Lee, K.-c., Anguelov, D., Sumengen, B. and Gokturk, S. B.: Markov random field models for hair and face segmentation, *Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 1–6 (2008).

[11] Julian, P., Dehais, C., Lauze, F., Charvillat, V., Bartoli, A. and Choukroun, A.: Automatic hair detection in the wild, *Proceedings of International Conference on Pattern Recognition*, pp. 4617–4620 (2010).

[12] : OpenCV. `http://opencv.org/`.

[13] 飯塚健男，和田俊和：色弁別度を用いた実時間ステレオ対象検出・追跡，画像の認識理解シンポジウム（MIRU），pp. 1072–1077 (2006).

[14] 飯塚健男：弁別度を用いた実時間ステレオ対象検出・追跡，和歌山大学大学院システム工学研究科修士論文 (2006).

[15] Wada, T.: Visual Object Tracking Using Positive and Negative Examples, *Robotics Research*, Springer, pp. 189–199 (2011).

[16] 大池洋史，呉海元，加藤丈和：鮮明な画像撮影のための高速追従カメラシステム，知能メカトロニクスワークショップ講演論文集，Vol. 9, pp. 79–84 (2004).

[17] Bradski, G. R.: Computer vision face tracking for use in a perceptual user interface, *Proceedings of IEEE Workshop on Applications of Computer Vision*, pp. 214–219 (1998).

[18] Oike, H., Wu, H. and Wada, T.: Adaptive selection of non-target cluster centers for k-means tracker, *Proceedings of International Conference on Pattern Recognition*, pp. 1–4 (2008).

[19] 大池洋史：能動カメラの高速追従制御による移動物体の鮮明な画像撮影方法に関する研究，和歌山大学大学院システム工学研究科博士論文 (2009).

[20] Qi, Y. and Wu, H.: A Pixel-wise tracking algorithm using stereo camera, *Proceedings of International Conference on Ubiquitous Robots and Ambient Intelligence*, pp. 198–201 (2010).

[21] Isard, M. and Blake, A.: ICONDENSATION: Unifying low-level and high-level tracking in a stochastic framework, Computer Vision—ECCV'98, Springer, pp. 893–908 (1998).