

# 修 士 論 文

## 滑動性眼球運動を用いた 視線計測の自動校正

2016年2月3日 提出

指導教員 佐藤 洋一 教授

情報理工学系研究科 電子情報学専攻

48-146447 松本 大輝

# 内容梗概

人の視線はその背後に潜む心的状態を推定する上で大きな手掛かりとなる情報であり，その推定技術（視線推定）はマーケティングなど多方面の分野に応用されている．既存の視線推定の多くは，ユーザがディスプレイに映るコンテンツを閲覧する状況を想定しており，ディスプレイ上の基準点と瞳孔中心・目領域の見えといった眼球情報の関係性を学習する手続き（キャリブレーション）が事前に必要となる．しかし，このような手続きはユーザごとに行う必要があるため，街中の電子公告など，ユーザが頻繁に切り替わる状況では適用できないといった問題がある．

そこで本研究では，ユーザへの指示を必要としない非明示的なキャリブレーション手法を提案する．提案手法では，特にユーザが映像を閲覧するような状況を想定し，滑動性眼球運動と呼ばれる動く対象を追視する際に発生する眼球運動に着目する．そして，映像中の動きと眼球運動との相関や，その動きの視覚的顕著性を利用することで，キャリブレーションに利用可能な基準点を自動で検出する．これにより，例えばテレビ番組を閲覧する際のユーザの視線を解析する場合に，テレビ番組の合間に流れるコマーシャル映像を利用してキャリブレーションを行うことでユーザへの指示なしに視線推定を行うことが可能になる．そして，公開データセット ASCMN を用いたキャリブレーションの精度評価実験の結果，比較手法に比べ精度が向上したことから提案手法の有効性が示された．

# 目次

第 1 章	序論	1
1.1	背景	1
1.2	目的とアプローチ	2
1.3	構成	3
第 2 章	関連研究	4
2.1	眼球運動との相関を用いたキャリブレーション手法	5
2.2	顕著性を用いたキャリブレーション手法	6
2.3	他ユーザの視線パターンを利用したキャリブレーション手法	7
2.4	パソコン上の操作を手掛かりとしたキャリブレーション手法	7
2.5	関連研究の比較と本研究の立ち位置	7
第 3 章	提案手法	9
3.1	問題設定	9
3.2	注視領域候補の生成	11
3.3	視線の階層構造の生成	12
3.4	注視領域候補の評価値の計算	13
3.5	整数計画問題による注視領域の推定	14
3.6	キャリブレーション	17
第 4 章	実験	19
4.1	擬似的な未校正注視点データを用いたキャリブレーション実験	19
4.1.1	実験設定	19
4.1.2	比較手法	20
4.1.3	実装	20
4.1.4	実験結果と考察	24
4.2	実際の未校正注視点データを用いたキャリブレーション実験	31

---

4.2.1	未校正の注視点データの取得 . . . . .	31
4.2.2	実験設定 . . . . .	33
4.2.3	実験結果と考察 . . . . .	34
第 5 章	結論 . . . . .	39
5.1	まとめ . . . . .	39
5.2	課題と今後の展望 . . . . .	39
5.2.1	未校正の注視点データの定義 . . . . .	39
5.2.2	注視領域候補の選択方法 . . . . .	40
5.2.3	オンライン処理によるキャリブレーション . . . . .	41
参考文献		42
発表文献		46

# 目次

1.1	一般的なキャリブレーション手順 . . . . .	2
2.1	眼球モデル . . . . .	5
2.2	顕著性マップの例 . . . . .	6
3.1	ディスプレイ上の基準点 (+) を注視しているときの推定した注視点 . . . . .	10
3.2	提案手法の流れ . . . . .	12
3.3	Dense trajectory の例 . . . . .	13
3.4	視線の階層構造の例 . . . . .	14
3.5	視線の階層構造の生成の流れ . . . . .	15
3.6	整数計画問題を用いる目的 . . . . .	16
4.1	実験で用いた映像の例 . . . . .	21
4.2	実験で用いなかった映像の例 . . . . .	22
4.3	視線の階層構造の各分節が固視であるか滑動性眼球運動であるかの判定例 . . . . .	22
4.4	Harel の手法 [11] を用いて生成した顕著性マップの例 . . . . .	23
4.5	各映像での提案手法を用いたキャリブレーション結果 . . . . .	25
4.6	キャリブレーションに適した映像の例 . . . . .	26
4.7	キャリブレーションに不向きな映像の例 . . . . .	27
4.8	提案手法と比較手法 1 との比較結果 . . . . .	28
4.9	比較手法 1 において選ばれたスケールに相当する視線の階層構造における階層 . . . . .	28
4.10	提案手法と比較手法 2 との比較結果 . . . . .	29
4.11	ユーザごとのキャリブレーション結果 . . . . .	30
4.12	注視点データを取得する際にディスプレイに表示する基準点の位置関係 . . . . .	32
4.13	キャリブレーション結果を取得する際にディスプレイに表示する基準点の位置関係 . . . . .	32
4.14	注視点とその周りの基準点の位置関係の例 . . . . .	32

4.15	ユーザの注視点データを取得する際の実験環境 . . . . .	33
4.16	未校正の注視点データの例 . . . . .	34
4.17	各映像での提案手法を用いたキャリブレーション結果 . . . . .	35
4.18	提案手法と比較手法 1 との比較結果 . . . . .	36
4.19	比較手法 1 において選ばれたスケールに相当する視線の階層構造における階層	36
4.20	提案手法と比較手法 2 との比較結果 . . . . .	37
4.21	ユーザごとのキャリブレーション結果 . . . . .	38
5.1	未校正の注視点データと真の注視点データとのエラーマップ . . . . .	40

# 表目次

2.1	関連研究の比較 . . . . .	8
-----	-------------------	---

# 第 1 章

## 序論

### 1.1 背景

視線推定とは，ユーザが閲覧コンテンツ，あるいは視界中のどこに目を向けているかを推定する技術<sup>\*1</sup>であり，広告での視線誘導 [22]，自動車の安全運転支援 [15]，注視に基づくインタフェース [14] といった様々な分野に応用されている．その基本的なアプローチは，ユーザがディスプレイに提示されたコンテンツを閲覧する際の眼球情報（瞳孔の位置や目領域の見え）をカメラで撮影し解析するというものであり，これまでに数々の手法が提案されている [10]．

多くの視線推定手法では，キャリブレーションと呼ばれる手続きが事前に必要となる．これは，図 1.1 のようにディスプレイに提示された複数の基準点を注視するようユーザに教示し，基準点の位置に応じて眼球情報がどのように変化するかを学習する手続きである．この手続きはユーザごとに必要となるため，ユーザが頻繁に切り替わる状況への適用は困難である．例えば，デパートの入口にそのデパート内の店の商品などが書かれた電子公告を設置し，人々がその電子公告を見ながら行きたい店や買いたい商品を探しているような状況を考える．このとき，電子公告の前に立ち止まる人は頻繁に切り替わるためユーザごとにキャリブレーションを行うことは困難である．しかし，キャリブレーションができたならば，ユーザの視線を解析し，そのユーザが興味を持ちそうな商品をさらに提示するといったユーザの好みに応じた広告の提示ができると考えられる．その他にも，例えばテレビ番組を閲覧するユーザの視線を推定する状況を考える．リビングなどでテレビを閲覧する際にはしばしば人の位置や姿勢が大きく変化するため頻繁にキャリブレーションが必要となる．そこで，テレビ番組の合間に流れるコマーシャル映像を利用してキャリブレーションを行うことができれば，姿勢等が変わるたびに基準点を注視する必要がなくなるといった利点がある．

---

<sup>\*1</sup> 本研究では，ユーザがディスプレイを閲覧する状況を想定し，その際のディスプレイ上の 2 次元位置を視線として扱う．ただし，[9, 12, 18] のように視線を 3 次元ベクトルとして考える場合もある．



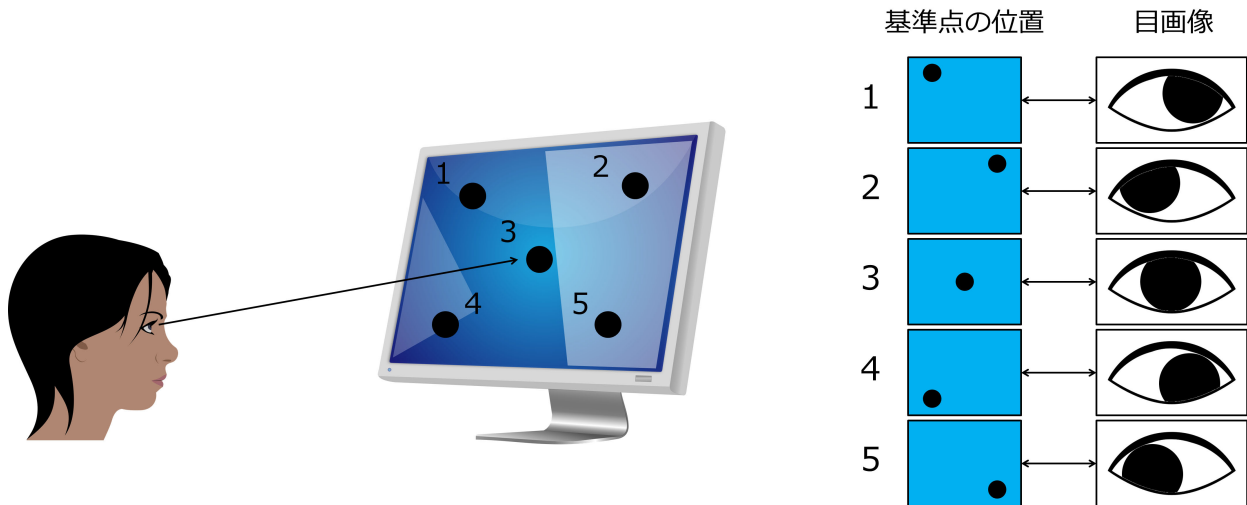


図 1.1 一般的なキャリブレーション手順. ディスプレイ上に提示された複数の基準点を注視するようにユーザに教示し, その際にカメラで撮影した目画像から得られる眼球情報と基準点の位置との対応からキャリブレーションを行う.

## 1.2 目的とアプローチ

本研究では, ディスプレイ上の基準点をユーザに注視させるといった明示的な指示をすることなく, キャリブレーションを行うことを目的とする. その一つのアプローチとして, ディスプレイに提示された映像をユーザが閲覧するような状況を想定し, このような非明示的なキャリブレーションをユーザが映像を閲覧する間に自動で行うような枠組みの構築を目指す. 本研究では, このキャリブレーションを行うにあたって, 映像を閲覧する際にしばしば発生する滑動性眼球運動に着目する. 滑動性眼球運動とは, 動く対象を追視する際に発生する眼球運動であり, この区間において映像中で動きが眼球運動と相関し, かつ目を引き付けやすい領域を注視領域として推定する. この領域を前述の基準点として利用することで, ユーザに特別な教示を与えることなくキャリブレーションを行うことができる.

関連研究として映像中の動きと眼球運動との相関を利用したキャリブレーション手法 [21] と顕著性を利用したキャリブレーション手法 [26, 5, 27, 6] の主に 2 種類がある. 眼球運動との相関を利用した手法では, ユーザが閲覧する映像をあらかじめデザインする必要がある上に, 一部のオブジェクトのみが動くような単純な映像のみを対象としている. これに対して, 本研究ではホームビデオや監視カメラの映像のような一般的な映像にも適用できるような手法を提案する. また, 顕著性を利用した手法は静止画ベースの手法である. これに対して, 本研究ではそれを映像へと拡張し, 動きの顕著性を利用する.

## 1.3 構成

本論文の構成を述べる。

第 1 章では本研究の研究背景，目的，アプローチを述べる。

第 2 章では本研究に関係する非明示的なキャリブレーション手法として眼球運動との相関を用いる手法，顕著性を用いる手法，他ユーザの視線パターンを用いる手法，パソコン上の操作を手掛かりとする手法の大きく 4 種類を紹介し，それらを比較した上で本研究の立ち位置を述べる。

第 3 章では本研究において提案する手法を説明する。提案手法では，映像から注視領域となる候補を多数生成し，整数計画問題の枠組みでその候補の中からキャリブレーションに適したものを注視領域として推定する。

第 4 章では評価実験による提案手法の有効性の検証について報告する。実験では，公開データセット ASCMN から擬似的に生成した注視点データと，実際にユーザが映像を閲覧する際に収集した注視点データの両方に対してキャリブレーションの精度評価実験を行った。

第 5 章では本研究の結論を述べる。

## 第 2 章

# 関連研究

視線推定手法は、補間に基づく手法、モデルベースの手法、アピランスペースの手法の大きく 3 種類に分類される [10]. 補間に基づく手法 [15, 19, 3, 4] では、目画像から得られる瞳孔の中心と角膜反射との位置関係といった特徴を引数とする多項式関数を構成し、その多項式関数を用いて注視点を推定する. そして、キャリブレーションを行うことによって多項式関数の係数を決定する. モデルベースの手法 [9, 12, 18] では、幾何学的な眼球モデルを用いて瞳孔の中心や角膜反射などの特徴から三次元的な視線方向を推定する. 図 2.1 に示す眼球モデルにおいて、視線方向とは視軸のことであり、この視軸を推定する際にはまず光軸を推定し、光軸と視軸の成す角度だけずらすことによって推定する. しかし、この光軸と視軸のなす角度は人によってばらつきがあるため [25], キャリブレーションによってこの角度を推定する. アピランスペースの手法 [2, 35, 29, 33] では、目画像そのものを高次元の特徴とみなし、その特徴を用いて注視点を推定する. 目画像と注視点との関係をあらかじめ学習する必要があるため、この学習データを得るためにキャリブレーションが必要となる.

このようにいずれの視線推定手法においてもキャリブレーションが必要不可欠である. キャリブレーションを行う際には、ユーザはディスプレイ上の複数の基準点を注視する必要がある. また、より簡単にキャリブレーションを行うために、ディスプレイ上の 1 点の基準点のみを注視することでキャリブレーションを行う手法 [20] も提案されている. しかしながら、例えばスーパーマーケットでの買い物や、工場現場における作業など、特定の活動に従事するユーザの視線を調査する場合を考えると、その活動を妨げることのない受動的な視線推定が必要となり、ディスプレイ上の基準点を注視するといった明示的なキャリブレーション操作は行えない.

そこで本研究では、明示的なキャリブレーション操作を行うことなく自動でキャリブレーションが行われるような手法の開発に取り組む. 非明示的なキャリブレーション手法として、眼球運動との相関を用いる手法、顕著性を用いる手法、他ユーザの視線パターンを利用する手法、パソコン上での操作を手掛かりとする手法の大きく 4 種類が挙げられる. これらについて

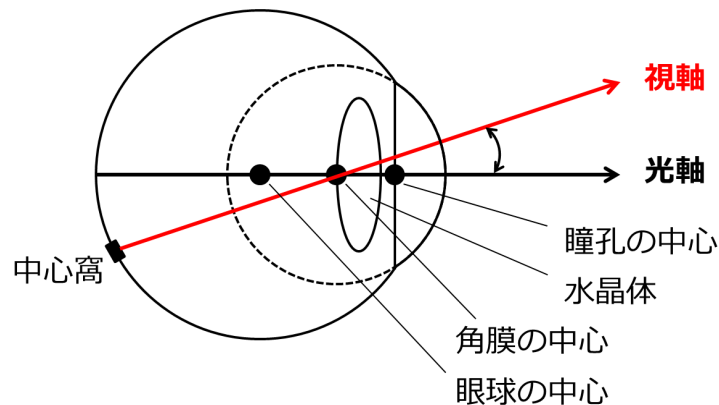


図 2.1 眼球モデル。視線とは視軸のことであり、視軸は中心窩と角膜の中心を通る直線である。モデルベースの手法では、光軸を推定し、光軸と視軸の成す角度だけずらすことによって視軸を推定する。

次に説明する。

## 2.1 眼球運動との相関を用いたキャリブレーション手法

映像中の動きと眼球運動との相関を用いたキャリブレーション手法として Pfeuffer らの手法 [21] が挙げられる。この手法では、ユーザが映像を閲覧する状況を想定し、映像中において眼球運動との相関が高いオブジェクトを注視オブジェクトとして推定することによってキャリブレーションを行う。しかし、この手法では閲覧する映像をあらかじめ新たにデザインする必要がある上に、一部のオブジェクトのみが動くような単純な映像のみを対象としている。そのため、監視カメラの映像やホームビデオのような一般的な映像において複数の領域が複雑な動きをしようる場合には適用できないといった問題がある。

また、映像中の動きと眼球運動との相関を用いた注視領域推定手法として [36, 30, 31, 7] が挙げられる。米谷ら [36] は、ユーザがディスプレイ上に提示された商品の画像といった複数のオブジェクトを閲覧しながら、どれを選択するかを考えているような状況を想定とした手法を提案している。ユーザに提示する各オブジェクトは動きを持っており、その動きと眼球運動との相関および同期の程度を分析することでいつどのオブジェクトが注視されたかを推定する。Vidal ら [30, 31] は、ユーザがディスプレイ前に近づくとき座にインタラクションを開始できるような、視線を入力としたインタフェースを提案している。Esteves ら [7] は、ユーザがスマートウォッチを使用している状況を想定し、スマートウォッチ上に円軌道上を動く点を複数表示させ、それを注視することで音楽の音量を調節するといった様々なインタラクションを行うことができるインタフェースを提案している。ここで表示する点は位置や回転方向や角速度などが互いに異なっており、その動きと眼球運動との相関からどの点を注視しているのかを推

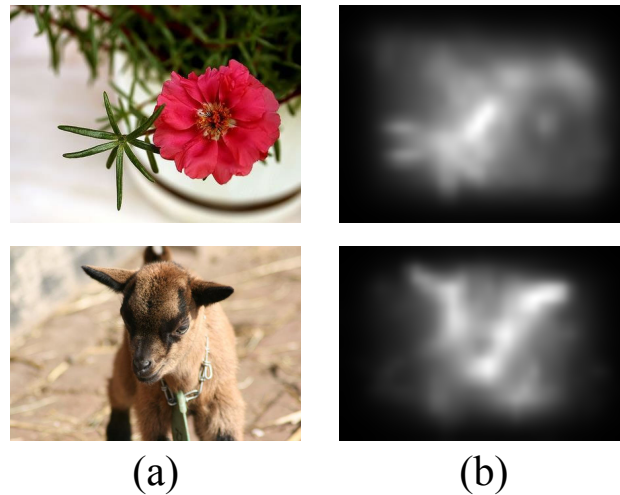


図 2.2 顕著性マップの例. (a) 入力画像. (b) 顕著性マップ.

定する.

## 2.2 顕著性を用いたキャリブレーション手法

人間は他の領域に比べ顕著な領域を注視しやすいため、映像の視覚的顕著性に基づくアプローチは注視領域を推定する上で有効である. 顕著性を用いたキャリブレーション手法として [26, 5, 27, 6] が挙げられる. 菅野ら [26, 27] は, ユーザが映像を閲覧する状況を想定とした手法を提案している. ここでは, 図 2.2 のような顕著性マップ [13] と呼ばれる, 映像中の各位置がどの程度視覚的注意を向けられやすいかを数値で表現したマップを用いて確率的に注視点を推定している. しかし, この手法では映像フレームの顕著性マップとそれを注視する際の目画像の組を学習データとしてあらかじめ大量に用意する必要がある上に, ユーザの頭部が動くと目領域の見えが変化するため頭部を固定する必要があるといった制約がある. また, この手法と類似した枠組みで精度を向上させた手法として Choi らの手法 [6] がある. 一方, Chen ら [5] は, ユーザが画像を閲覧する状況を想定とした手法を提案している. ここでは, モデルベースの視線推定手法をベースとし, 顕著性マップを用いてキャリブレーションにおいて必要な視軸と光軸との成す角度を確率的に推定している. この手法では事前に学習データを用意する必要がないため, このシステムを即座に適用できるといった利点がある.

## 2.3 他ユーザの視線パターンを利用したキャリブレーション手法

他ユーザの視線パターンを利用したキャリブレーション手法として Alnajar らの手法 [1] が挙げられる。ここでは、人間は同じコンテンツを閲覧する際には同じような領域を注視するという仮定を置き、他ユーザの注視点位置を手掛かりにユーザの注視点を推定している。しかし、他ユーザの平均的な視線パターンを得るために学習データとしてコンテンツを閲覧する際の複数人の注視点情報を用意する必要があるといった制約がある。

## 2.4 パソコン上の操作を手掛かりとしたキャリブレーション手法

パソコン上の操作を手掛かりとしたキャリブレーション手法として菅野らの手法 [28] が挙げられる。ここでは、ユーザがマウスをクリックする際にはマウスカーソルに視線が向けられているという仮定を置き、マウスがクリックされたときのマウスカーソルの位置とそのときに得られた目画像との組からキャリブレーションを行っている。しかし、この手法ではパソコンでの操作、特にマウスをクリックする操作が必要となるため、利用用途が制限されてしまうといった問題がある。

## 2.5 関連研究の比較と本研究の立ち位置

これまで紹介したキャリブレーション手法を比較したものを表 2.1 に示す。これらの手法において、キャリブレーションを行う際にユーザが閲覧するコンテンツは基本的には映像か画像のどちらかになる。眼球運動には、人間の心的状態に基づく内因性の眼球運動と、外環境の視覚刺激などによる外因性の眼球運動の 2 種類があり、コンテンツを閲覧する際の眼球運動には両者が混在する。内因性の眼球運動は閲覧するコンテンツからは推定できないため、キャリブレーションを行う際には、心的状態に影響されない閲覧するコンテンツによる外因性の眼球運動のみを用いることが望ましい。これを踏まえた上で、画像のように静止している対象を注視する場合には、注視対象の切り替えにおいて内因性の眼球運動が起きる。これに対して映像のように動いている対象を注視する場合には、その動きに沿って視線を動かすとき、つまり滑動性眼球運動を行っているとき、外因性の眼球運動のみとなる。したがって、キャリブレーションを行う際にユーザが閲覧するコンテンツは画像よりも映像の方が望ましい。また、映像中の動きと眼球運動との対応が取れたとき、その動きの各フレームでの位置がキャリブレーション

表 2.1 関連研究の比較

キャリブレーション手法	眼球運動との相関 [21]	顕著性 [26, 27]	顕著性 [5]	他者の視線パターン [1]	パソコン操作 [28]
閲覧するコンテンツ	映像	映像	画像	画像	デスクトップ
学習データ	不要	必要	不要	必要	不要

を行う際のディスプレイ上の基準点として利用することができる。そのため、一般的なキャリブレーション手法のようにユーザがディスプレイ上の複数の基準点を注視する場合よりも多くの基準点と眼球情報との対応関係を得ることができ、より精度よくキャリブレーションを行えることが期待できる。また、実用的にはディスプレイとユーザの目を撮影するカメラさえ用意すれば即座に視線を推定できる方が良く、それを考慮するとキャリブレーションにあたって事前の学習データは Pfeuffer らや Chen らの手法のように不要である方が望ましい。

これらを踏まえ、本研究では Pfeuffer らの手法のようにユーザが映像を閲覧する状況を想定する。そして、注視領域を推定する際には眼球運動との相関に加え、動きの顕著性も考慮する。これによって推定する領域を人間が視覚的注意を向けやすい領域のみに絞り込むことでより高精度の注視領域推定を行い、キャリブレーションを行う。そして、本研究では Pfeuffer らの手法では適用できなかった一般的な映像への適用を目指す。これにあたって大きく課題が二つある。まず、一つ目の課題は注視領域候補を映像中からどのように見つけ評価するのかという点である。一般的な映像では映像中の動きは未知であるため、映像中にどのような動きが現れるかを推定する必要がある。また、一般的な映像では複数の領域が類似した動きをしまう場合があるため、このとき眼球運動との相関だけでは注視領域を一意に推定することは困難である。そこで、映像から多数の注視領域候補を生成することで映像中の動きを捉え、各注視領域候補に対して眼球運動との相関だけでなく顕著性も組み合わせて注視領域らしさを評価することでこの課題を解決する。そして、二つ目の課題はキャリブレーションに適した滑動性眼球運動をどのように検出するかという点である。閲覧する映像が異なると視線の動きのスケールが変化するため、適切な長さの滑動性眼球運動を抽出することは困難である。そこで、様々な長さの滑動性眼球運動の候補を生成し、その中から適切な滑動性眼球運動の候補を選ぶ問題に置き換えることでこの課題を解決する。

## 第 3 章

# 提案手法

### 3.1 問題設定

ユーザが映像を閲覧するという状況を想定し，眼球運動から非明示的にキャリブレーションを行う．そのために，Alnajjar らの手法 [1] と同様に，真の注視点データが何らかの形で変換された（例えば線形変換）未校正の注視点データを入力として想定する．すなわち，本研究におけるキャリブレーションとは，未校正の注視点データから真の注視点データへの変換パラメータを推定することに相当する．

このような未校正の注視点データは，例えば事前に獲得された他ユーザのキャリブレーション結果を用いた視線推定によって得ることが可能である．図 3.1 にディスプレイ上の基準点を注視しているときの推定した注視点を示す．赤点が本人のキャリブレーション結果を基に推定した注視点，青点が他ユーザのキャリブレーション結果を基に推定した注視点を表しており，青点のように真の注視点に比べ多少ずれたものを入力とする．また，本研究では [1] と同様に，視線推定において最も一般的な変換であるスケールとバイアスに焦点を当てるために，ここでの未校正の注視点データは真の注視点データに対してスケールとバイアスのみがかかっているような状況を想定する．

特に本研究では，入力映像として複数の領域が同じ動きをしようる一般的な映像，例えば，監視カメラの映像のように複数の人物が同一方向に歩行する映像や，ホームビデオのようにあるイベントをカメラのパン・チルトを行いながら撮影した映像にも適用可能な枠組みの提案を目指す．このとき，キャリブレーション問題を解くにあたって，以下の課題がある．

■課題 1. 注視領域候補をどう見つけ評価するか 従来研究 [21] ではあらかじめ映像をデザインするため映像中のオブジェクトの動きは既知であった．これに対して，本研究では既存の映像を用いるため映像中のオブジェクトの動きは未知である．そのため，映像中にどのような動きが現れるかを推定する必要がある．また，従来研究が対象としていた映像では複数の領域が



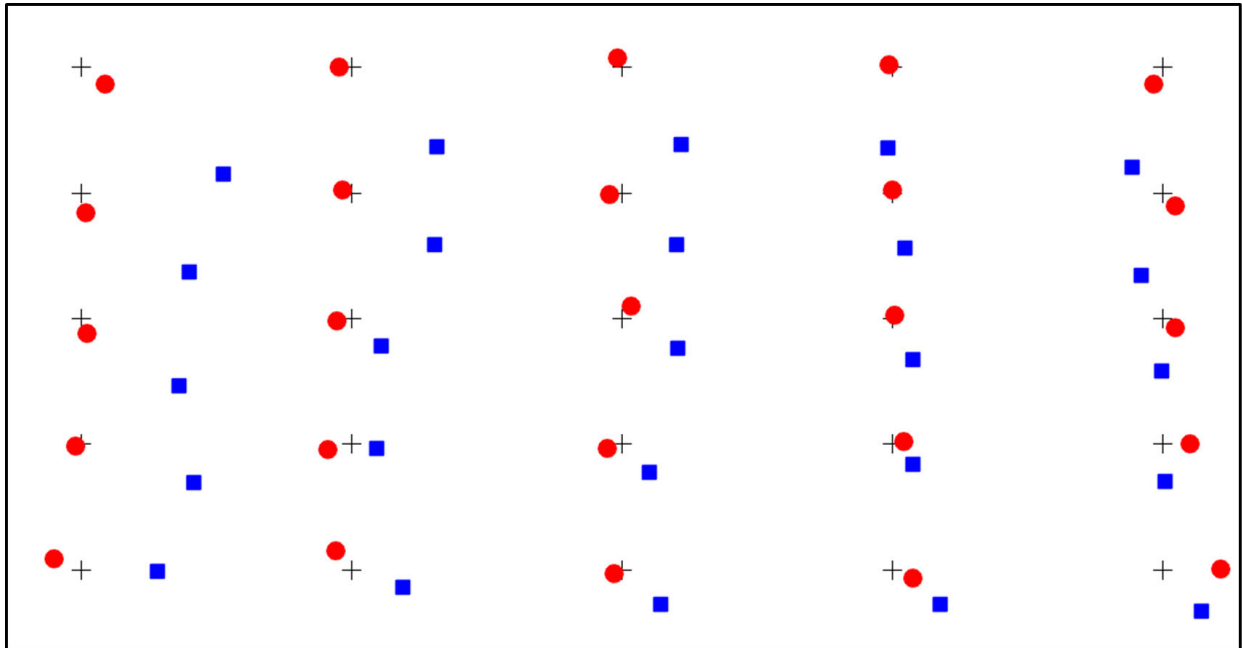


図 3.1 ディスプレイ上の基準点 (+) を注視しているときの推定した注視点. 赤点は本人のキャリブレーション結果を基に推定した注視点, 青点は他ユーザのキャリブレーション結果を基に推定した注視点を表している. 青点のように真の注視点に比べ多少ずれた注視点を未校正の注視点データとする.

類似する動きをしなかったのに対し, 本研究が対象とする一般的な映像では複数の領域が類似する動きをする場合が存在する. このとき, 眼球運動との相関のみを用いて注視領域を一意に定めることは困難である.

■課題 2. キャリブレーションに適した滑動性眼球運動の区間をどう見つけるか 映像中の動きと眼球運動との相関を求める際には, ユーザが実際に動くオブジェクトを追視している区間, つまり滑動性眼球運動を行っている区間を検出する必要がある. 滑動性眼球運動は, 従来手法 [16, 17] ではまずサッケードと呼ばれる高速な眼球運動を検出し, 次に残りのフレームに対して滑動性眼球運動であるか固視 (fixation) と呼ばれる一定の領域内に視線が留まる運動であるかを判定することによって検出される. つまり, サッケードを正しく検出できなかった場合, 例えば本来は長い滑動性眼球運動の区間であったものが間にサッケードが誤って検出されてしまい短く分断されるといったように, 滑動性眼球運動の検出にも影響が現れる. これに加えて, サッケードの検出においては, 従来手法 [16] では隣接フレーム間の速度といったように, 何らかの尺度に対して閾値を設定して検出する手法が用いられることが多い. この閾値は真の注視点データに対して正解ラベルがつけられた学習データに基づいて決められる. しかし, 本研究では入力として未校正の注視点データを使用する上に, 閲覧する映像が異なると視線の動きのスケールも変化するため, このような学習ベースの手法を用いて単一の閾値を設定

し、サッケードを検出することは困難である。

これら 2 つの課題に対し、本研究では以下のようなアプローチにより解決を図る。まず、映像中から多数の特徴点を追跡することで注視領域候補を生成し、各注視領域候補に対して眼球運動との相関だけでなく顕著性も組み合わせて注視領域らしさを評価することによって課題 1 を解決する。また課題 2 を解決するために、サッケードを検出する際の閾値を複数個設定し、各閾値を用いて注視点データを複数の時区間に分割することで、様々な長さの滑動性眼球運動の候補を生成する。これらを実現する提案手法の流れ (図 3.2) は以下ようになる。まず、映像中の動きを捉えるために、入力映像から多数の特徴点を抽出し、それらを追跡することによって注視領域候補を生成する (図 3.2(a))。次に、様々な長さの滑動性眼球運動の候補を抽出するために、サッケードを検出する際の閾値を複数個設定し、各閾値を用いて入力注視点データを複数の時区間に分割することによって、視線の階層構造を生成する (図 3.2(b))。得られた階層構造の各分節において、その分節内に存在する注視領域候補に対して視線の動きとの相関およびその候補の動きの顕著性に基づき注視領域らしさを評価する (図 3.2(c))。顕著性も考慮することで、注視領域候補を人が視覚的注意を向けやすい領域のみに絞ることができる。例えば、カメラのパン・チルトが含まれる映像の場合、背景領域は眼球運動との相関は高くなるものの顕著性は低くなるため候補から除外することができる。そして、キャリブレーションに適した滑動性眼球運動の候補と注視領域候補を選択する問題を整数計画問題に定式化し、それを最適化することによって注視領域を推定する (図 3.2(d))。最後に、推定した注視領域を用いてキャリブレーションを行う (図 3.2(e))。また、キャリブレーションは水平方向、垂直方向それぞれ独立に行う。このキャリブレーション結果を基に入力の注視点データを変換したものを出力とする。それぞれのステップに対して詳しく説明する。

## 3.2 注視領域候補の生成

映像中から眼球運動と相関するような領域を推定するためには、まず映像中にどのような動きが現れるかを推定し、注視領域候補を生成する必要がある。注視領域候補の生成には dense trajectory [32] を用いる。ここでは、映像中に現れる多数の特徴点を Good features to track [24] と呼ばれる、自己相関行列の最小固有値が閾値を超えるかどうかという基準で抽出し、オプティカルフロー [8] と呼ばれる映像中の各画素についての 1 フレーム先における変位量を用いてその特徴点を長時間追跡したものを注視領域候補とする。

図 3.3 に dense trajectory の例を示す。図 3.3 のように人間や自動車といった動く領域に対してその動きを追跡できていることが分かる。

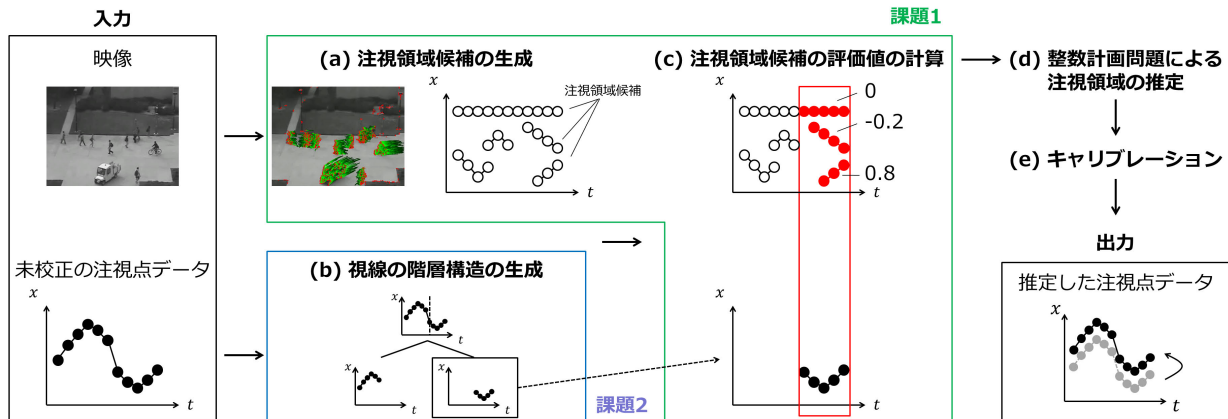


図 3.2 提案手法の流れ. 入力映像とそれを閲覧する際の未校正の注視点データを使用する. 提案手法は以下の 5 ステップから構成されている. まず, 映像中から多数の注視領域候補を生成する (a). そして, 様々な長さの滑動性眼球運動候補を抽出するために入力の注視点データから視線の階層構造を生成する (b). 滑動性眼球運動の各候補内に存在する各注視領域候補に対して注視領域らしさを表す評価値を計算する (c). キャリブレーションに適した滑動性眼球運動候補と注視領域候補を推定する問題を整数計画問題で定式化し最適化を行う (d). 最後に, 推定した注視領域を用いてキャリブレーションを行う (e). このキャリブレーション結果を基に入力の注視点データを変換したものを出力とする.

### 3.3 視線の階層構造の生成

映像中の動きと眼球運動との相関を求める際には, ユーザーが実際に滑動性眼球運動を行っている区間を検出する必要がある. しかし, 滑動性眼球運動を検出するには, まずサッケードを検出する必要があり, その際に必要となる閾値を適切に設定することは困難であるという課題があった. そこで, 本研究ではサッケードを検出する際の閾値を複数個設定し, 入力の注視点データを複数の分節に分割することによって様々な長さの滑動性眼球運動の候補を生成する.

入力の注視点データから様々な長さの滑動性眼球運動の候補を生成するため, 本研究では多重解像度解析 [34] を用いたアプローチ [37] を利用して視線の階層構造を生成する. まず,  $t$  フレーム目の注視位置を  $(x_t, y_t)$  としたときに, 隣接フレーム間の視線の移動速度  $\mu_t = \sqrt{(x_{t+1} - x_t)^2 + (y_{t+1} - y_t)^2}$  を計算する. 得られた速度信号  $\mu = (\mu_1, \mu_2, \dots, \mu_N)$  を視線の階層構造を生成する際の入力信号とする (図 3.5(a)).

信号  $\mu$  に対してスケール  $\sigma_1, \sigma_2, \dots, \sigma_K (\sigma_k < \sigma_{k+1})$  のガウス関数を畳み込むことによって, 信号  $\mu$  のスケール空間を構築することができる. このとき, 得られた波形集合においてスケールを変化させながら波形の 1 次微分のゼロ交差 (変曲点) を追跡することで, ガウス関数の因果性により, スケールが小さくなるにつれて新たな変曲点が現れるような階層構

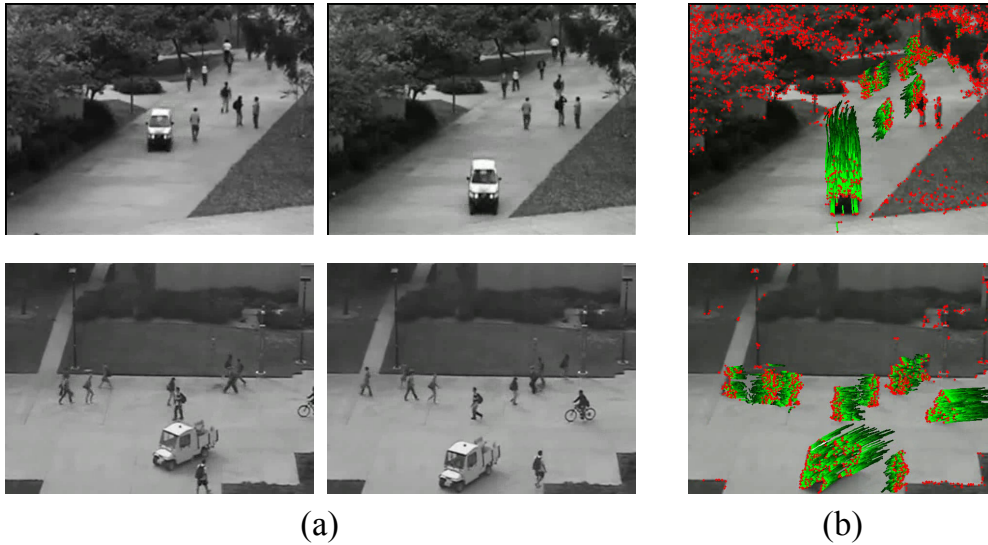


図 3.3 Dense trajectory の例. (a) 映像フレーム. (b) Dense trajectory. 赤点は特徴点, 緑線はその特徴点の軌跡を表している.

造が形成される (図 3.5(b), (c)). ここでの波形の 1 次微分は視線の加速度であり, その変曲点は速度の極大点に相当するため, これをサッケードとみなすというアプローチをとる. このとき, この変曲点をスケールを変化させながら追跡することによってサッケードを検出することができる. ここでは,  $\sigma_k \rightarrow \sigma_{k-1} (k = 2, 3, \dots, K)$  のスケールの変化により必ず新たな変曲点が現れ, スケール  $\sigma_K$  では変曲点を持たないようにスケールを設定する. そして, 各スケールにおいて新たに現れた変曲点を追跡したときに得られるスケール  $\sigma_1$  での変曲点をそのスケールにおける視線の分割点とする (図 3.5(d)). この分割点によって分割された区間を滑動性眼球運動の候補区間とする. これにより, スケールが小さくなるにつれてより細かく分節化されるような視線の階層構造 (図 3.4) が得られる.

この視線の階層構造の各区間には, 滑動性眼球運動あるいは固視が含まれる. ここでは滑動性眼球運動に着目するため, 明らかに固視を行っている分節を階層構造から除外する. ここでは, 各分節内における注視点がある一定の領域内に収まっている場合, 具体的には注視点の水平方向と垂直方向それぞれの最大値と最小値の差分の二乗和の平方根 ( $\sqrt{\{\max(x) - \min(x)\}^2 + \{\max(y) - \min(y)\}^2}$ ) が閾値以下の場合に固視と判定する.

### 3.4 注視領域候補の評価値の計算

3.3 節で得られた階層構造の各分節において, その分節内に存在する注視領域候補に対して注視領域らしさを評価する. ここでは, 眼球運動との相関と自身の顕著性がともに高いものを注視領域として考える. なぜならば, 動いている領域を追視する場合, その領域の動きと眼球運動は相関し, かつ背景領域と比べ動きを持っているため自身の動きの顕著性も高いためであ

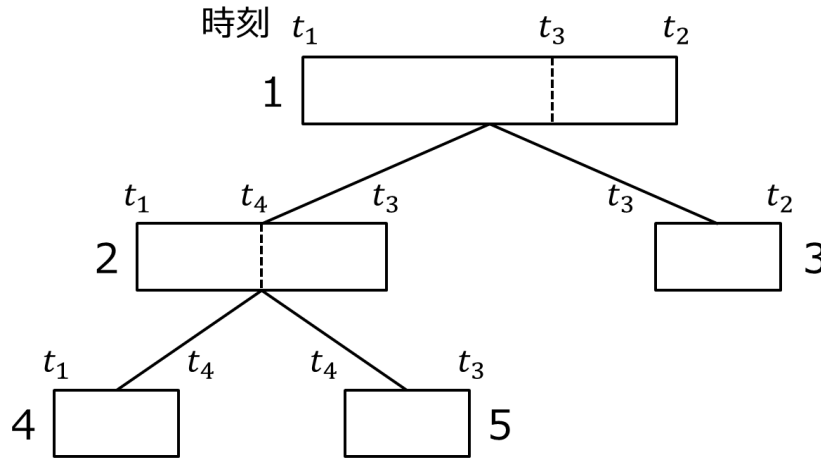


図 3.4 視線の階層構造の例. それぞれの分節が滑動性眼球運動の候補区間に相当する. 分節 1 は時刻  $t_3$  を分割点として分節 2 と分節 3 に分割され, 分節 2 は時刻  $t_4$  を分割点として分節 4 と分節 5 に分割されている.

る. そのため, ここでの評価値は注視領域候補の動きと眼球運動との相関と, 注視領域候補の動きの顕著性を掛け合わせたものと定義する. 相関は正規化相互相関とし, 顕著性は注視領域候補の各フレームでの位置の顕著性マップの値を平均したものとする. 具体的には, 注視点系列を  $\mathbf{x} = (x_1, x_2, \dots, x_n), \mathbf{y} = (y_1, y_2, \dots, y_n)$  (ただし,  $\mathbf{x}$  は水平方向,  $\mathbf{y}$  は垂直方向を表す), 注視領域候補の軌跡を  $\mathbf{v} = (v_1, v_2, \dots, v_n), \mathbf{w} = (w_1, w_2, \dots, w_n)$  (ただし,  $\mathbf{v}$  は水平方向,  $\mathbf{w}$  は垂直方向を表す), フレーム  $t$ , 位置  $(x, y)$  での顕著性マップの値を  $S_t(x, y) \in \mathbb{R}$  とすると, 評価値  $e_x, e_y$  は次のように表される.

$$e_x = \frac{\sum_i (x_i - \bar{x})(v_i - \bar{v})}{\sqrt{\sum_i (x_i - \bar{x})^2} \sqrt{\sum_i (v_i - \bar{v})^2}} \times \frac{1}{n} \sum_i S_i(v_i, w_i) \quad (3.1)$$

$$e_y = \frac{\sum_i (y_i - \bar{y})(w_i - \bar{w})}{\sqrt{\sum_i (y_i - \bar{y})^2} \sqrt{\sum_i (w_i - \bar{w})^2}} \times \frac{1}{n} \sum_i S_i(v_i, w_i) \quad (3.2)$$

ここでは, 未校正の注視点データは真の注視点データに対してスケールとバイアスのみが水平方向と垂直方向それぞれ独立にかかっているような状況を想定しているため, 注視領域候補の動きと眼球運動との相関としてスケールとバイアスの変換に対応可能な正規化相互相関を用いている.

### 3.5 整数計画問題による注視領域の推定

図 3.6 に示すようにこれまでの手続きで, 入力 of 注視点データは階層的に分節化されており, 各分節には複数の注視領域候補がある. そして, 各注視領域候補にはその分節での視線との相関および自身の顕著性に基づいて注視領域らしさを表す評価値が与えられている. キャリ

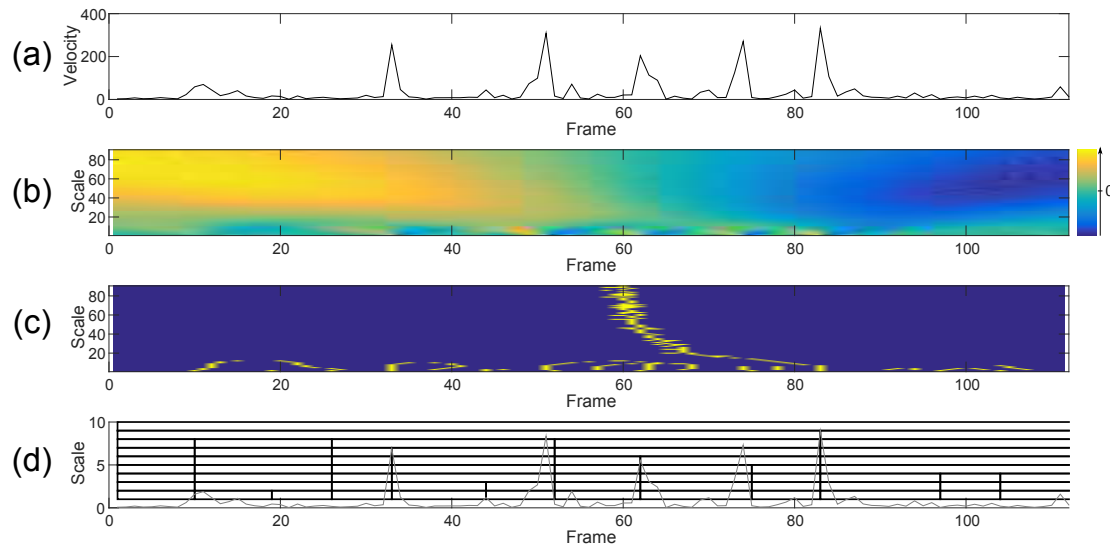


図 3.5 視線の階層構造の生成の流れ. 階層構造を生成する際の入力として隣接フレーム間の視線の移動速度の絶対値を求める (a). この入力に対して複数のスケールのガウス関数を畳み込むことによって波形集合を生成し, それぞれの波形の一次微分を計算する (b). 図中の色は一次微分の大きさに基づいている. 次に, この一次微分の変曲点をスケールを変化させながら追跡する (c). 図中の黄色の点の変曲点に対応する. そして, 各スケールにおいて新たに表れた変曲点を追跡したときに得られる最小スケールでの変曲点をそのスケールでの視線の分割点として, 階層構造を生成する (d). 図中において垂直方向に引かれた直線が視線の分割点に対応し, スケールが小さくなるにつれて細かく分節化される.

ブレイションのためには, ここから滑動性眼球運動候補および注視領域候補を選ぶ必要がある. ただし, その候補を選ぶ上で各時刻においてユーザはたかだか 1 個の領域候補を注視するという制約を満たしている必要がある.

これを同時に最適化するため, 本研究では整数計画問題で滑動性眼球運動候補と注視領域候補を選ぶ問題を定式化する. 基本的なアイデアは, 各時刻においてユーザはたかだか 1 個の領域候補を注視するという制約を満たしつつ, 選んだ注視領域候補の評価値の和が最大となるような選び方をするというものである. ここで, 各時刻においてユーザはたかだか 1 個の領域候補を注視するということを考慮するために, 選んだ滑動性眼球運動候補同士が時間的に重複しないといった制約を入れる. これによって, 選ばれる注視領域候補は選んだ各滑動性眼球運動候補において最大の評価値を持つものとなる. また, 選んだ注視領域候補の評価値の和が最大となるような選び方をした場合, 基本的に多くの分節を選んだ方が評価値の和が大きくなるため, 階層構造において下の層にある分節が選ばれやすくなり, 選ばれる分節に偏りが生じる. そのため, 例えば適切な長さの滑動性眼球運動候補が階層構造において上の層にあった場合には選ばれなくなるといった問題が生じる. そこで, どの分節も均等に選ばれるようにする

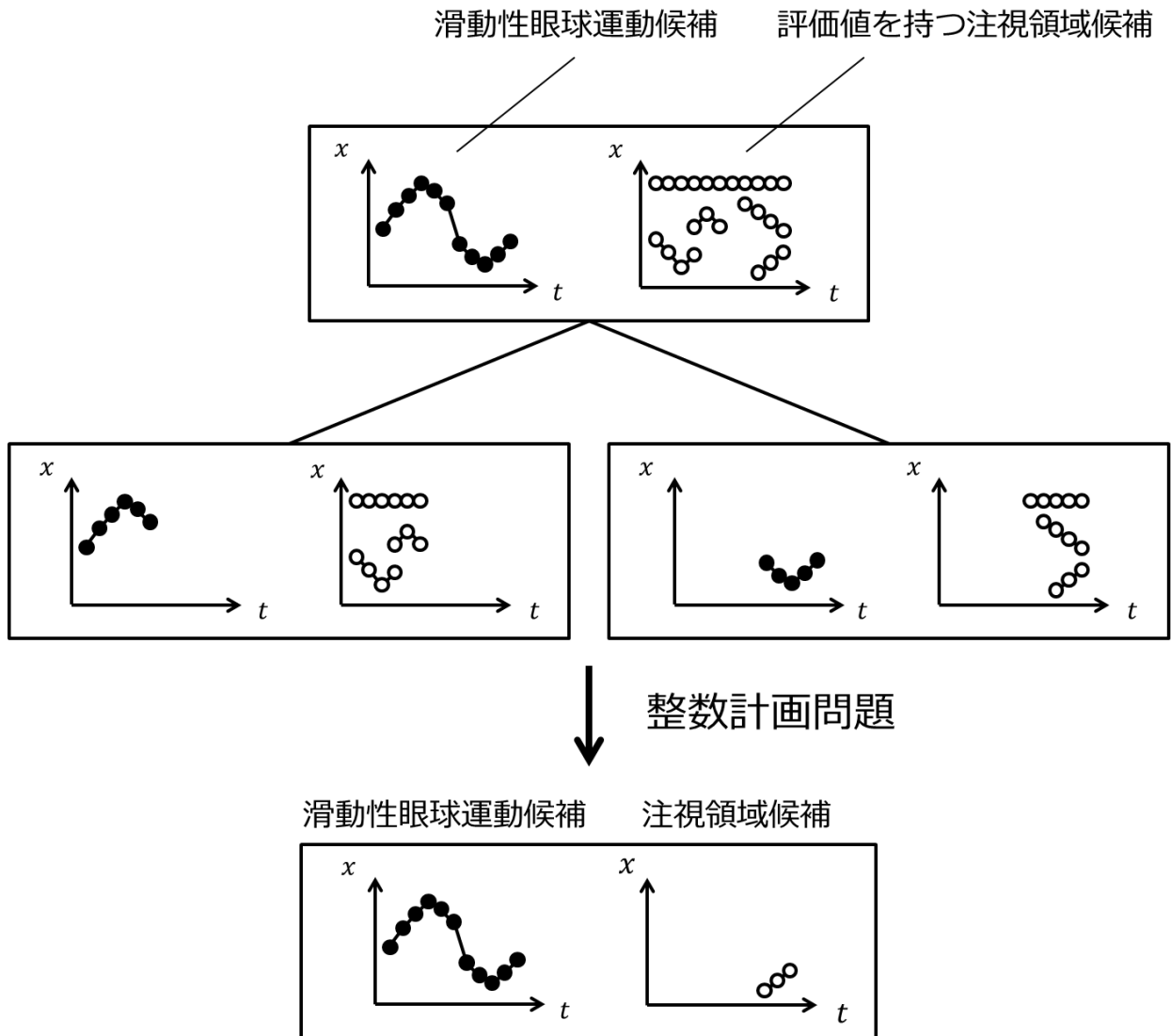


図 3.6 整数計画問題を用いる目的. これまでの手続きで, 入力 of 注視点データは階層的に分節化され, 各分節には注視領域らしさを表す評価値を持った注視領域候補がある. キャリブレーションのためには, ここから滑動性眼球運動候補と注視領域候補を選ぶ必要があり, この選択問題を整数計画問題で定式化することで最適化を行う.

ために, 階層構造において上の層にある分節であるほど注視領域候補の評価値が大きくなるような重みを加える. ここでは, 各分節が自身の下の層で最終的に何個に分節化されているかを調べ, その個数をその分節の重みとする.

具体的にはまず, ある  $i$  番目の分節が選ばれるかどうかを変数  $s_i \in \{0, 1\}$  で定義する. そして, 変数  $\lambda_i$  を分節  $i$  における注視領域候補の評価値の最大値と定義すると, 例えば視線の階層構造が図 3.4 のように表される場合には次式のような整数計画問題で滑動性眼球運動候補

と注視領域候補を選ぶ問題を定式化することができる。

$$\max_{\mathbf{s} \in \{0,1\}} \boldsymbol{\lambda}^T \mathbf{s} \quad s.t. \quad C\mathbf{s} = \mathbf{1} \quad (3.3)$$

$$\mathbf{s} = \begin{pmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \end{pmatrix}, \boldsymbol{\lambda} = \begin{pmatrix} 3\lambda_1 \\ 2\lambda_2 \\ \lambda_3 \\ \lambda_4 \\ \lambda_5 \end{pmatrix}, C = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \end{pmatrix}$$

図 3.4 において分節 1 は下の層で分節 3, 4, 5 の 3 個の分節に分節化されているため、評価値  $\lambda_1$  に対して重み 3 がかけられている。同様に、評価値  $\lambda_2$  に対しては重み 2 がかけられている。また、制約条件では、選ばれる分節が時間的に重複しない、つまり図 3.4 では分節の選び方が  $\{1\}, \{2, 3\}, \{3, 4, 5\}$  の 3 種類のみになるように行列  $C$  を設定する。このとき、制約条件を満たす  $\mathbf{s}$  は、 $\mathbf{s} = (1, 0, 0, 0, 0)^T, (0, 1, 1, 0, 0)^T, (0, 0, 1, 1, 1)^T$  となるので、 $3\lambda_1$  と  $2\lambda_2 + \lambda_3$  と  $\lambda_3 + \lambda_4 + \lambda_5$  の大小関係から選択する滑動性眼球運動候補の組を決定し、注視領域を推定する。

### 3.6 キャリブレーション

3.5 節で推定した注視領域とその時刻における注視点データの組を用いてキャリブレーションを行う。本研究では、入力の注視点データはスケールとバイアスが未校正であるという前提を置いているため、キャリブレーションでは真の注視点データに対してかけられたスケールとバイアスの変換パラメタを最小二乗法を用いて推定する。ここでは、評価値が高いほどその注視領域が注視されている可能性が高いと仮定し、評価値を重みとした重み付き最小二乗法を用いる。具体的には、注視点系列を  $\mathbf{x} = (x_1, x_2, \dots, x_n), \mathbf{y} = (y_1, y_2, \dots, y_n)$  (ただし、 $\mathbf{x}$  は水平方向、 $\mathbf{y}$  は垂直方向を表す)、注視領域の軌跡を  $\mathbf{v} = (v_1, v_2, \dots, v_n), \mathbf{w} = (w_1, w_2, \dots, w_n)$  (ただし、 $\mathbf{v}$  は水平方向、 $\mathbf{w}$  は垂直方向を表す)、注視領域の評価値を  $\mathbf{e}_x = (e_{x1}, e_{x2}, \dots, e_{xn}), \mathbf{e}_y = (e_{y1}, e_{y2}, \dots, e_{yn})$  (ただし、 $\mathbf{e}_x$  は水平方向の評価値、 $\mathbf{e}_y$  は垂直方向の評価値を表す) とすると、変換パラメタ  $(a_x, b_x, a_y, b_y)$  は次式の最小化問題を解くことによって得られる。

$$E_x = \sum_i^n e_{xi}(v_i - a_x x_i - b_x)^2 \quad (3.4)$$

$$E_y = \sum_i^n e_{yi}(w_i - a_y y_i - b_y)^2 \quad (3.5)$$



つまり，水平方向では  $\frac{\partial E_x}{\partial a_x} = 0, \frac{\partial E_x}{\partial b_x} = 0$  を解くことによって次式のように求められる．

$$\begin{pmatrix} a_x \\ b_x \end{pmatrix} = \left( \left( \begin{pmatrix} e_{x1}x_1 & e_{x1} \\ e_{x2}x_2 & e_{x2} \\ \vdots & \vdots \\ e_{xn}x_n & e_{xn} \end{pmatrix}^T \begin{pmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{pmatrix} \right)^{-1} \begin{pmatrix} e_{x1}x_1 & e_{x1} \\ e_{x2}x_2 & e_{x2} \\ \vdots & \vdots \\ e_{xn}x_n & e_{xn} \end{pmatrix}^T \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix} \right) \quad (3.6)$$

同様にして，垂直方向では  $\frac{\partial E_y}{\partial a_y} = 0, \frac{\partial E_y}{\partial b_y} = 0$  を解くことによって次式のように求められる．

$$\begin{pmatrix} a_y \\ b_y \end{pmatrix} = \left( \left( \begin{pmatrix} e_{y1}y_1 & e_{y1} \\ e_{y2}y_2 & e_{y2} \\ \vdots & \vdots \\ e_{yn}y_n & e_{yn} \end{pmatrix}^T \begin{pmatrix} y_1 & 1 \\ y_2 & 1 \\ \vdots & \vdots \\ y_n & 1 \end{pmatrix} \right)^{-1} \begin{pmatrix} e_{y1}y_1 & e_{y1} \\ e_{y2}y_2 & e_{y2} \\ \vdots & \vdots \\ e_{yn}y_n & e_{yn} \end{pmatrix}^T \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{pmatrix} \right) \quad (3.7)$$

## 第 4 章

# 実験

提案手法の有効性を検証するため、二種類の実験を行った。まず、公開データセット ASCMN [23] にあるキャリブレーション済みの注視点データを基に擬似的に生成した未校正の注視点データを入力とした実験を行った。次に、実際にユーザが映像を閲覧する際の未校正の注視点データを集め、それを入力とした実験を行った。

### 4.1 擬似的な未校正注視点データを用いたキャリブレーション実験

#### 4.1.1 実験設定

公開データセット ASCMN [23] から 14 種類の映像\*1 (図 4.1) を用いて、キャリブレーションの精度評価を行った。ここでは、図 4.1(a) のように複数の人物が同一方向に歩行するといった一般にキャリブレーションが困難であると思われる映像に加え、図 4.1(b),(c) のように一部の領域のみが動くといったキャリブレーションが比較的容易であると見込まれる映像も使用した。一方、このデータセットには、図 4.2 のような人が密集しているような映像もあるが、このような映像は人が視覚的注意を向けやすい領域が画面全体にあるため提案手法の適用が難しいと判断し、本実験では除外した。

このデータセットには各映像に対して 10 人分のキャリブレーション済みの注視点データがあるため、本実験では入力の注視点データとして、キャリブレーション済みの注視点データに対して線形変換を施し、擬似的に未校正の注視点データを生成した。このとき、入力の注視点データ  $(x, y)$  と真の注視点データ  $(x', y')$  との関係は  $x = ax' + b, y = ay' + b$  と表され、この係数  $a$  と  $b$  をキャリブレーションによって推定する。本実験では、 $(a, b) = (2, 200)$  とした。

---

\*1 フレームレートは 15fps. 10 秒程度の短い映像が 6 種類, 30 秒から 1 分程度の長い映像が 8 種類.

精度評価は、推定された係数を基に入力の注視点データを変換したものと真の注視点データとの平方平均二乗誤差 (RMSE) を各映像のフレームサイズに対する割合に換算した値に基づいて行った。

### 4.1.2 比較手法

提案手法の有効性を検証するために、本研究では以下の 2 種類の比較手法を実装し評価した。

比較手法 1 特定のスケールで注視点データを分節化した場合。

視線の階層構造の生成 (3.3 節) において用いたスケール  $\sigma_1, \sigma_2, \dots, \sigma_K$  の中から 1 個のスケールを選び、そのスケールにおいて得られた視線の分割点によって分割された区間を滑動性眼球運動の区間とした。このとき、キャリブレーションにおいて用いる注視領域は、その滑動性眼球運動の各区間において最大の評価値を持つ注視領域候補となる。

比較手法 2 注視領域候補の評価値が眼球運動との相関のみの場合。

提案手法では、3.4 節で述べたように注視領域候補の評価値を眼球運動との相関と顕著性を掛け合わせたものと定義したが、比較手法ではこの顕著性の項を取り除き、注視領域候補の軌跡と眼球運動との正規化相互相関のみを注視領域候補の評価値として用いた。

提案手法と比較手法 1 を比較することで視線の階層構造を用いることの有効性を検証し、提案手法と比較手法 2 を比較することで顕著性を用いることの有効性を検証した。

### 4.1.3 実装

3.3 節での視線の階層構造の各分節が固視であるかの判定は、入力の注視点データを平均 0、分散 1 になるように正規化した状態で行い、判定の際の閾値は 0.3 とした。この判定手法を入力注視点データに適用した結果を図 4.3 に示す。図 4.3 から、視線の動きがほとんど見られない分節は固視と判定されていることが分かる。

また、3.4 節での顕著性マップの生成には Harel らの手法 [11]<sup>\*2</sup>を用いた。入力の映像から生成した顕著性マップを図 4.4 に示す。図 4.4 から、動いている領域ほど顕著性マップの値が高くなっていることが分かる。

そして、3.5 節での整数計画問題の解法には CPLEX<sup>\*3</sup>のライブラリを用いた。

<sup>\*2</sup> <http://www.vision.caltech.edu/~harel/share/gbvs.php>

<sup>\*3</sup> <http://www-03.ibm.com/software/products/ja/ibmilogcplex>

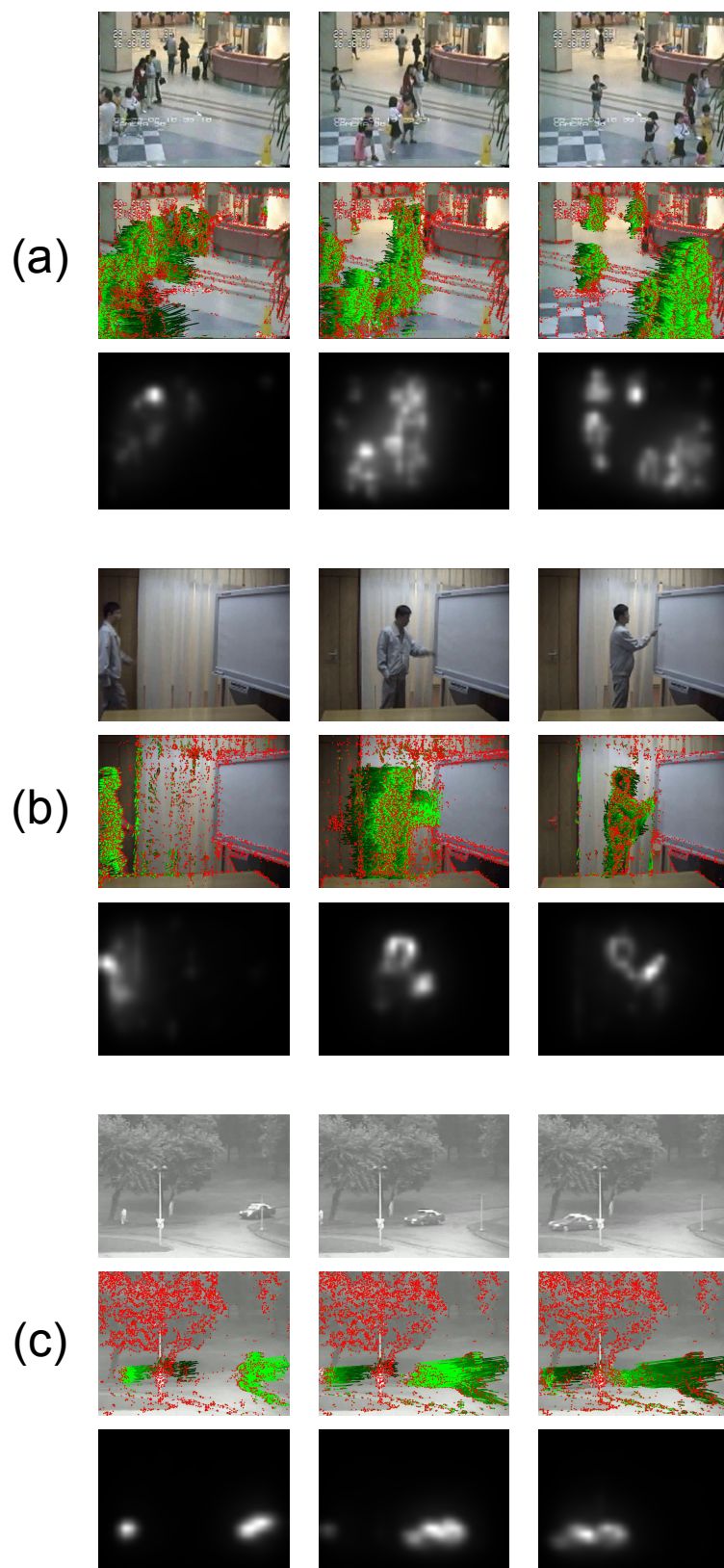


図 4.1 実験で用いた映像の例. (a) 建物内に設置された監視カメラの映像. (b) 男性がホワイトボード前へ歩行する映像. (c) 道端に設置された監視カメラの映像. それぞれ上段に映像フレーム, 中段に Dense trajectory, 下段に顕著性マップを示している.



図 4.2 実験で用いなかった映像の例。群衆を上から撮影した映像を表している。

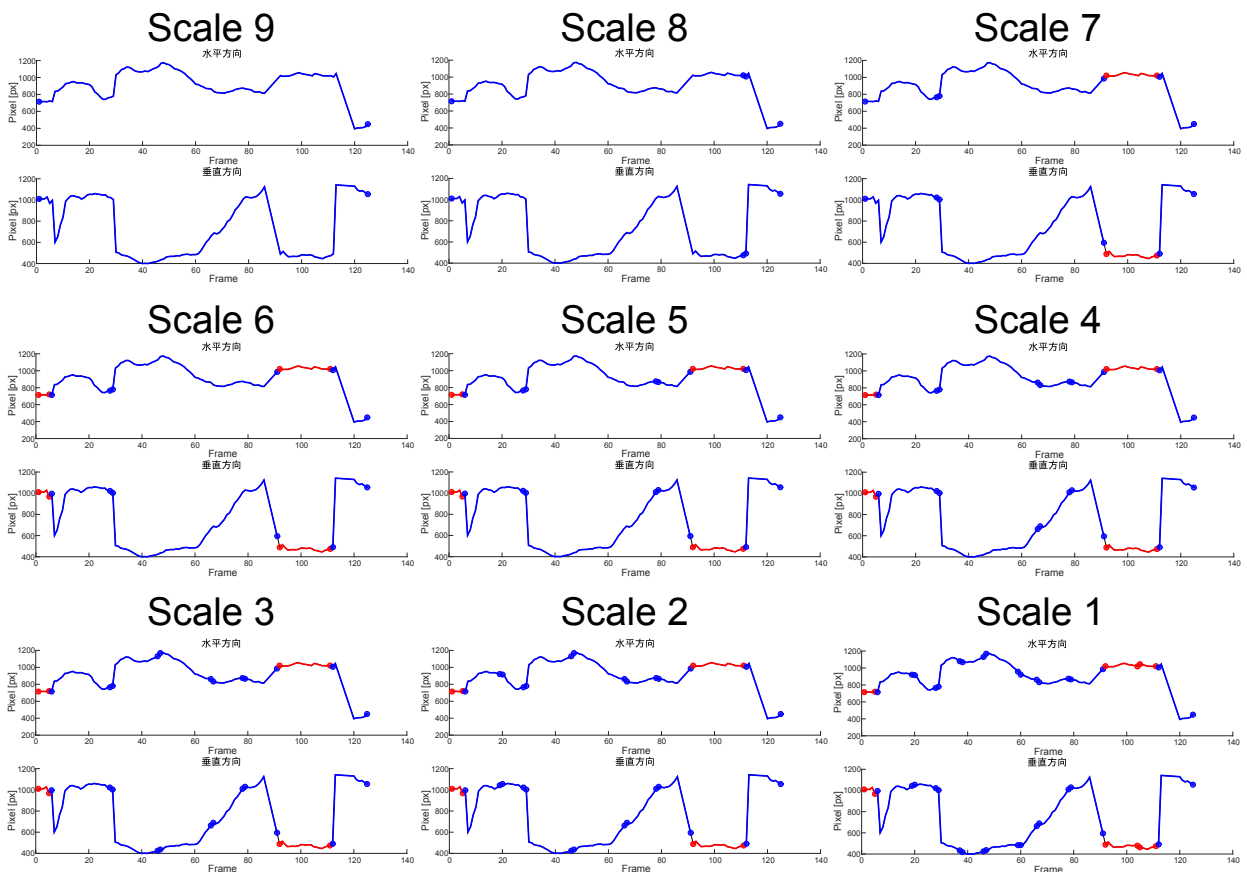


図 4.3 視線の階層構造の各分節が固視であるか滑動性眼球運動であるかの判定例。スケールごとに入力の注視点データが複数の分節に分割された結果を表しており、各グラフにおいて上段が水平方向、下段が垂直方向の注視点データを表している。また、青点で囲まれた分節は滑動性眼球運動、赤点で囲まれた分節は固視と判定されたことを表している。

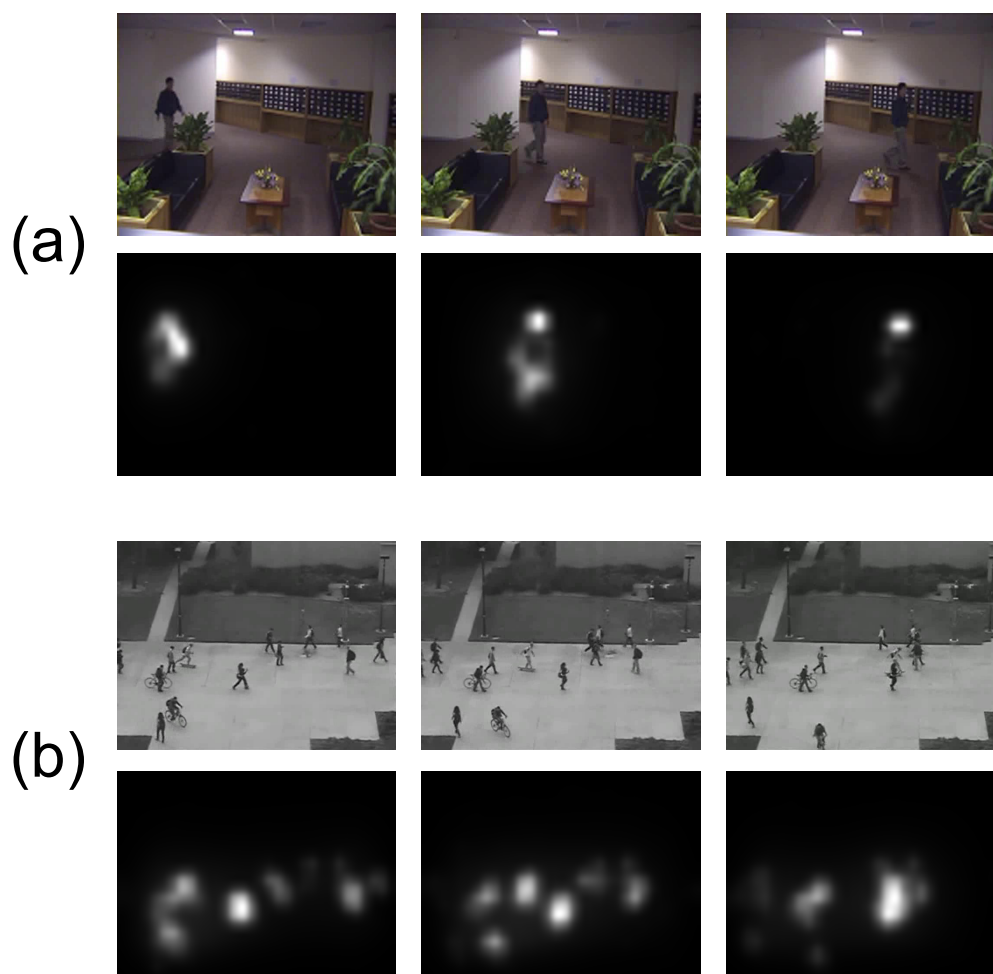


図 4.4 Harel の手法 [11] を用いて生成した顕著性マップの例. (a) 屋内に設置された監視カメラの映像. (b) 屋外に設置された監視カメラの映像. それぞれ上段に映像フレーム, 下段に顕著性マップを示しており, 顕著性マップにおいて色が白い領域であるほど視覚的注意を向けやすいことを表す.

#### 4.1.4 実験結果と考察

まず，図 4.5 に各映像での提案手法を用いたキャリブレーション結果を示す．図 4.5 から分かるように，映像 S1（図 4.6(a)）や S6（図 4.6(b)）のときにはキャリブレーションの精度が良い．これは，映像 S6 では 1 人の人物だけが歩行するといったように，これらの映像では同時刻において他と比べて顕著な動きが 1 個しかないため，それを注視しやすくなり，その結果として眼球運動との相関と顕著性からそれを注視領域として推定することが容易であったことが原因であると考えられる．一方，映像 L3（図 4.7(a)）や L4（図 4.7(b)）のときにはキャリブレーションの精度が悪いことが分かる．これは，提案手法では注視しているオブジェクトを推定できたとしても，そのオブジェクト全体が同じフローで動いている場合にはそのオブジェクト内のどこを注視しているかまでは特定できないということが原因の一つであると考えられる．例えば，映像 L3 のように歩行する人物を注視する場合，その領域内で得られる注視領域候補の動きはほぼ同一であると考えられるため，垂直方向においては最大で頭部から足までの大きな推定誤差が出る可能性がある．また，映像 L4 のように複数人が固まって同じ方向へ移動する場合にも同様に大きな推定誤差が出る可能性がある．そのため，このような映像はキャリブレーションには不向きであることが分かる．

図 4.8 に提案手法と比較手法 1 との比較結果を示す．比較手法 1 での結果は，複数のスケールで実験を行った際に精度が最も良かったスケールと最も悪かったスケールをデータごとに手動で選択し，それぞれを平均した結果を示している．図 4.8 から提案手法よりも比較手法 1 において最も精度の良いスケールを選択した方が精度が良いことが分かる．しかし，比較手法 1 において選択されたスケールが視線の階層構造において何層目にあたるのか（図 4.9）を見ると，ユーザごとに大きなばらつきがあることが確認できる．このことから単一のスケールを設定し視線を適切に分節化することは困難であることが実証でき，視線の階層構造を用いた方が精度は若干劣るものの妥当であることが分かる．次に，図 4.10 に提案手法と比較手法 2 との比較結果を示す．図 4.10 から比較手法 2 よりも提案手法の方が水平方向，垂直方向ともに精度が良いことが分かり，顕著性を用いることの有効性が確認できる．

図 4.11 に具体的なキャリブレーション結果の一例を示す．図 4.11 の上段は正しくキャリブレーションできた結果を表している．この映像では，複数の人物が歩行している中で，途中に 1 台の自動車が走行する．このように他と比べ目立った動きをする領域が 1 つだけの場合には，その領域だけを注視領域と推定し，キャリブレーションを行っていることが分かる．また，図 4.11 の下段は，キャリブレーションに一部失敗した結果を表している．この映像では，複数の人物が歩行している中で，途中に数台の自転車が走行する．このように複数の時区間で目立った動きが見られる場合には，水平方向での結果のように複数の区間を滑動性眼球運動とし，その中から注視領域を推定していることが分かる．一方，垂直方向での結果では赤枠部分

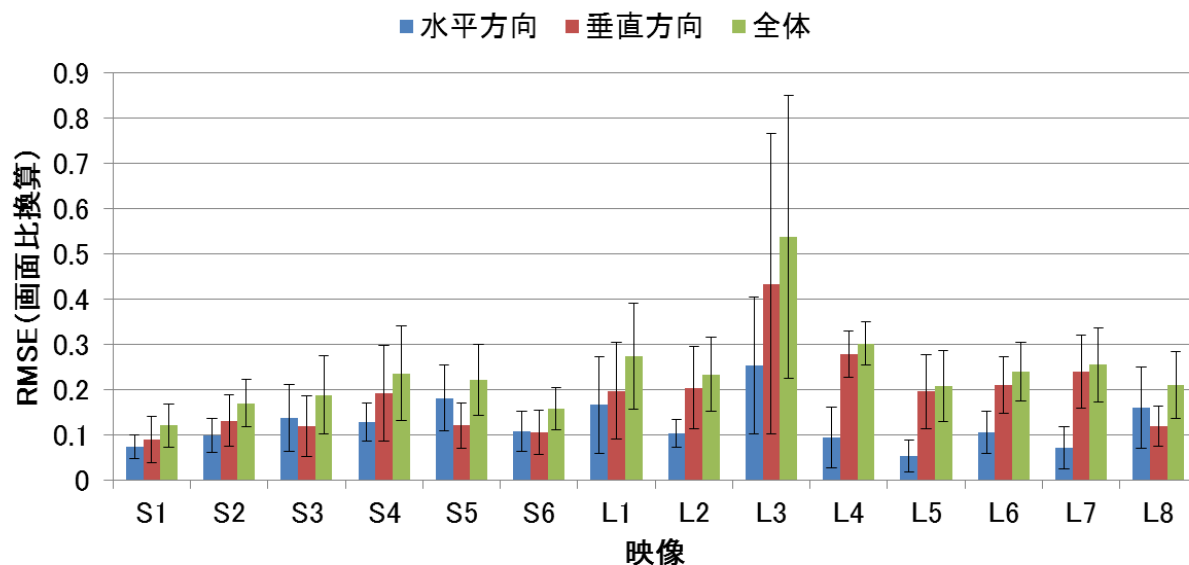


図 4.5 各映像での提案手法を用いたキャリブレーション結果. S1 から S6 が短い映像, L1 から L8 が長い映像を表している.

のように真の視線に比べ速度が全く異なる注視領域が選ばれていることが分かる. これは, 注視領域候補の評価値で用いている眼球運動との相関は正規化相互相関であるため, 注視領域候補の動きと眼球運動の方向さえ合っていれば相関が高くなり全体の評価値が高くなってしまふことが原因であると考えられる. これを解決するには, このような動きの速度も考慮するように注視領域候補の選び方を工夫する必要がある.



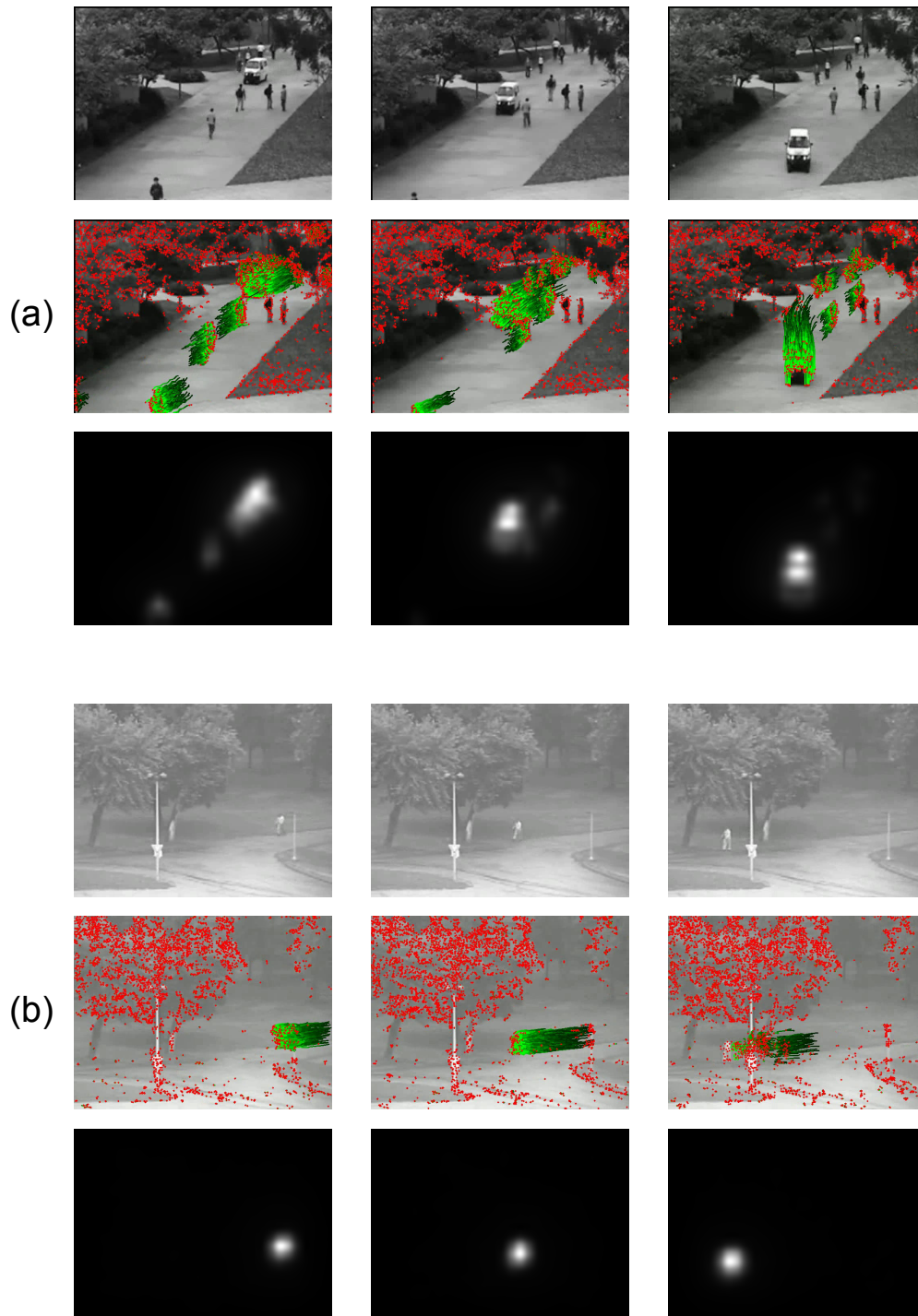


図 4.6 キャリブレーションに適した映像の例. (a) では一台の自動車が行走している. (b) では 1 人の人物が歩行している. また, それぞれ上段に映像フレーム, 中段に Dense trajectory, 下段に顕著性マップを示している.

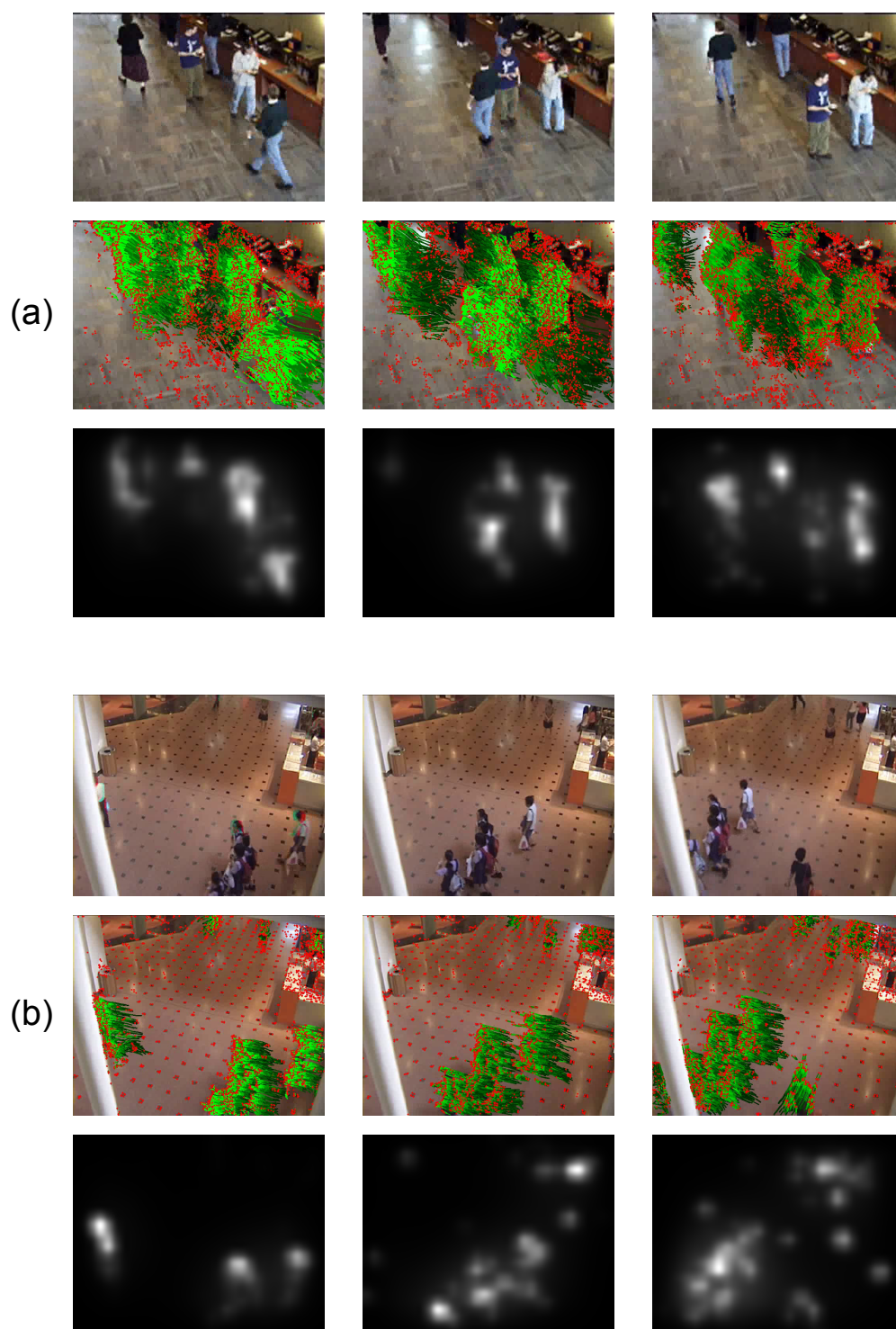


図 4.7 キャリブレーションに不向きな映像の例。(a) はレストラン内の監視カメラの映像であり、歩行する人が画面内に比較的大きく映っている。(b) は建物内の監視カメラの映像であり、複数人が固まって同じ方向へ移動している。また、それぞれ上段に映像フレーム、中段に Dense trajectory, 下段に顕著性マップを示している。

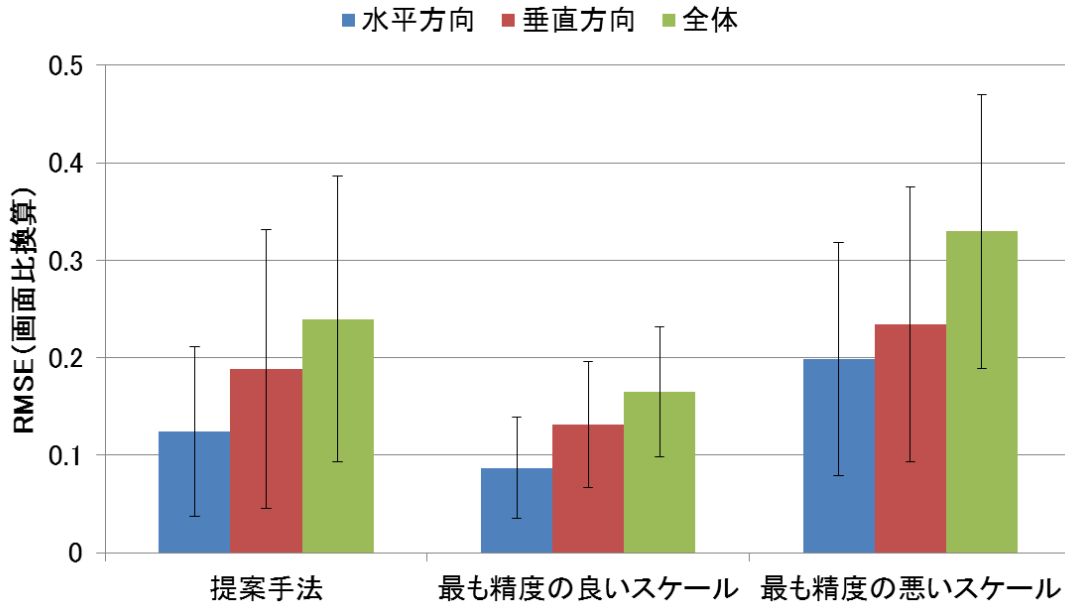


図 4.8 提案手法と比較手法 1 との比較結果. 比較手法 1 では, 複数のスケールで実験を行い, その中で最も精度の良かったスケールと最も精度の悪かったスケールをデータごとにそれぞれ手動で選択した際の結果を表している.

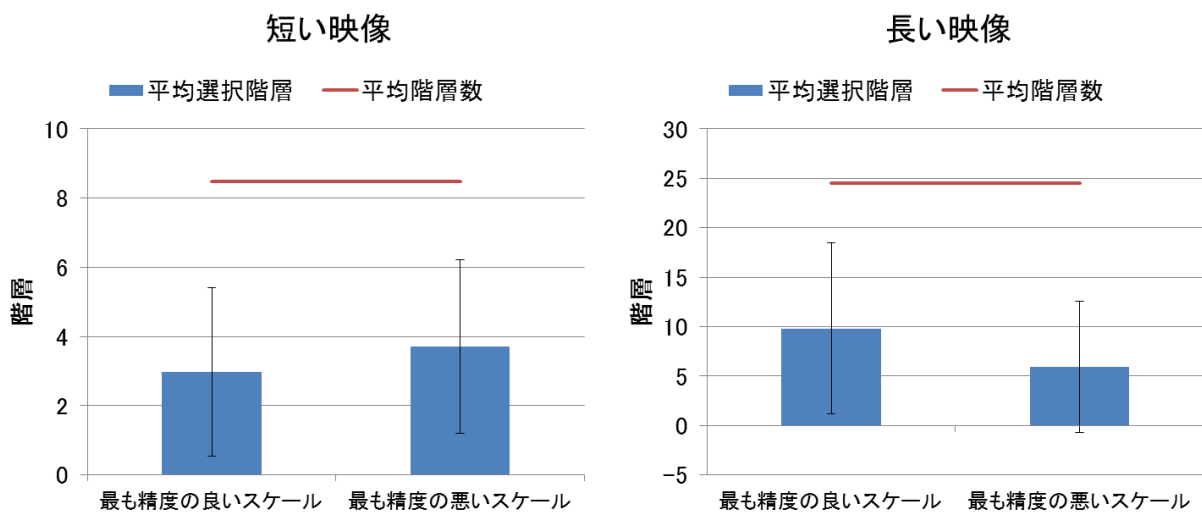


図 4.9 比較手法 1 において選ばれたスケールに相当する視線の階層構造における階層. 左が短い映像での結果, 右が長い映像での結果を表している.

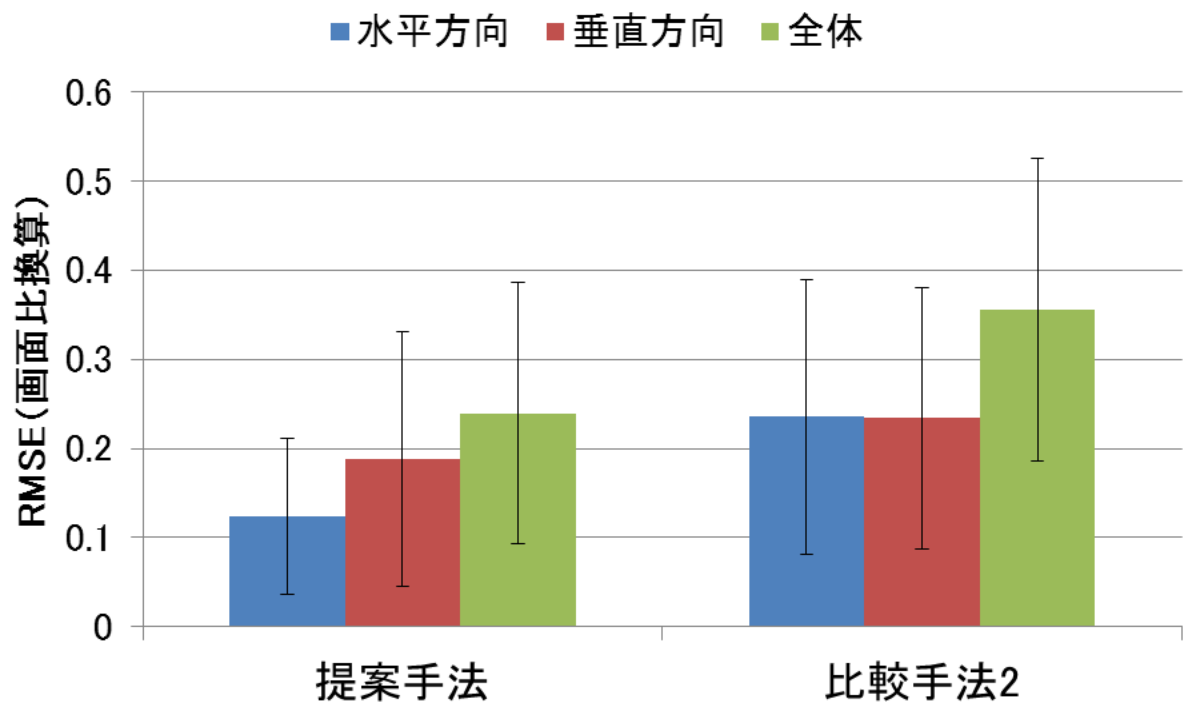
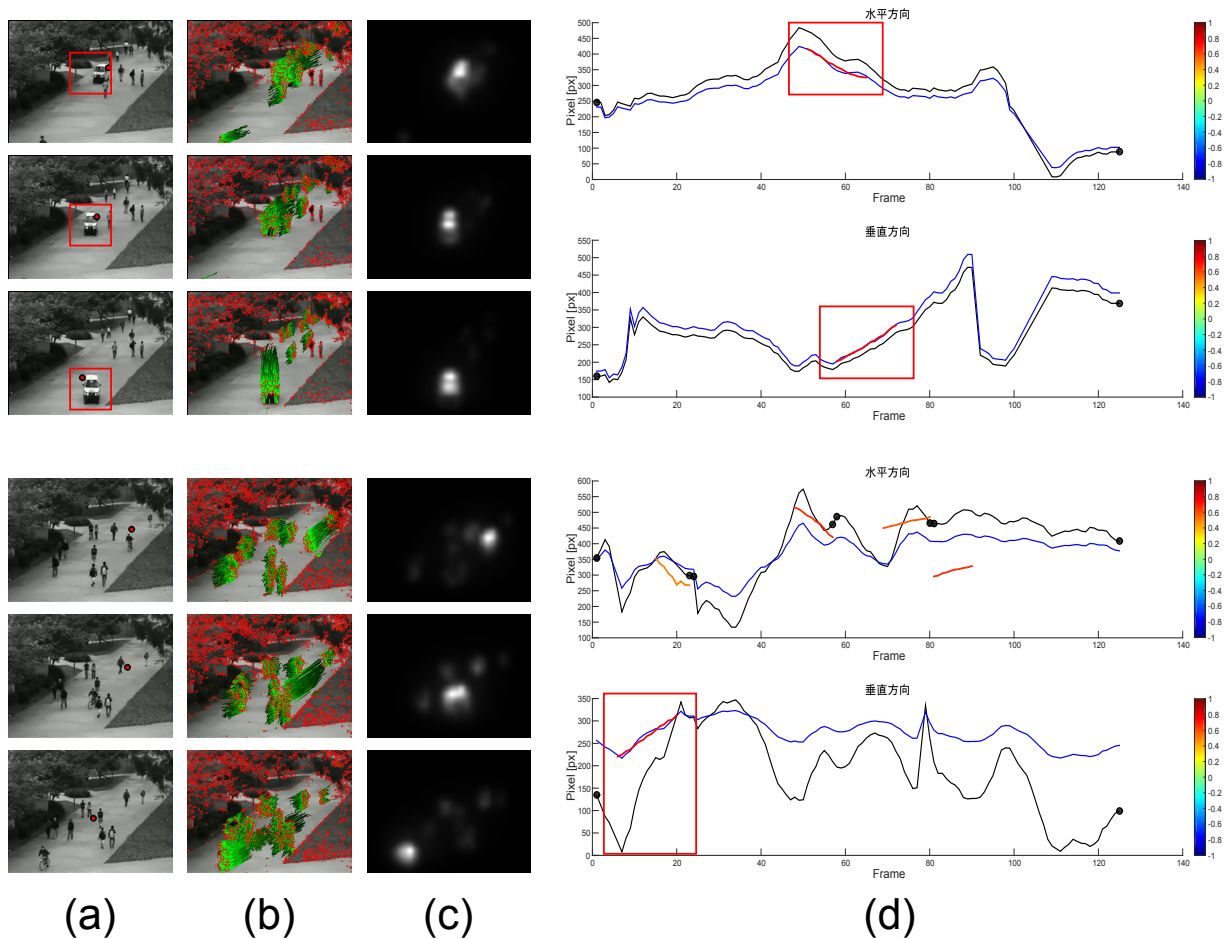


図 4.10 提案手法と比較手法 2 との比較結果



(a) (b) (c) (d)

図 4.11 ユーザごとのキャリブレーション結果. (a) 映像の数フレーム. フレーム内の赤点は真の注視点を表している. (b)Dense trajectory. (c) 顕著性マップ. (d) キャリブレーション結果. グラフは上段が水平方向での結果, 下段が垂直方向での結果を表している. また, 黒線が真の注視点データ, 青線は推定した変換パラメタを基に変換した注視点データ, 色のついた短線はキャリブレーションに用いた注視領域を表している. グラフ内での注視領域の色は評価値の大きさに基づいており, 評価値が高いほど赤く, 低いほど青くなる. そして, 黒点に囲まれた時区間が整数計画問題を解くことによって選ばれた分節を表している.

## 4.2 実際の未校正注視点データを用いたキャリブレーション 実験

### 4.2.1 未校正の注視点データの取得

実際にユーザが映像を閲覧する際の未校正の注視点データを取得するために、次のような手順を取った。まず、ディスプレイに図 4.12 のような 25 点の基準点を順に表示し、それを注視しているときの注視点データを取得する。この操作を 2 回繰り返す。1 回目では本人のキャリブレーション結果を基に注視点を推定し、2 回目ではあらかじめ獲得した他ユーザのキャリブレーション結果を基に注視点を推定する。この 2 種類の注視点データから、各基準点における注視点同士の誤差を計算する。次に、ディスプレイ全体に映像を表示し、それを閲覧する際の注視点データを本人のキャリブレーション結果を基に推定する。ここで得られた注視点データを、本実験では真の注視点データとして扱う。さらに、先ほどの各基準点での誤差を基に各注視点での誤差を推定し、真の注視点データに加えることによって未校正の注視点データを取得する。例えば、注視点  $P = (x, y)$  とその周りの 4 点の基準点  $P_i$  が図 4.14 のように位置しており、各基準点にはキャリブレーション結果の違いによる誤差  $\epsilon_i = (\alpha_i, \beta_i) \in \mathbb{R}^2$  が求められているとする。このとき、注視点での誤差  $\epsilon = (\alpha, \beta) \in \mathbb{R}^2$  をバイリニア補間によって推定する。まず、点  $Q_1 = (x, q_1)$  での誤差  $\epsilon_{Q_1} = (\alpha_{Q_1}, \beta_{Q_1}) \in \mathbb{R}^2$  を 2 点  $P_1, P_4$  での誤差から線形補間によって次のように求める。

$$\alpha_{Q_1} = \frac{p_2 - x}{p_2 - p_1} \alpha_1 + \frac{x - p_1}{p_2 - p_1} \alpha_4 \quad (4.1)$$

$$\beta_{Q_1} = \frac{p_2 - x}{p_2 - p_1} \beta_1 + \frac{x - p_1}{p_2 - p_1} \beta_4 \quad (4.2)$$

同様に、点  $Q_2 = (x, q_2)$  での誤差  $\epsilon_{Q_2} = (\alpha_{Q_2}, \beta_{Q_2}) \in \mathbb{R}^2$  も次のように求める。

$$\alpha_{Q_2} = \frac{p_2 - x}{p_2 - p_1} \alpha_2 + \frac{x - p_1}{p_2 - p_1} \alpha_3 \quad (4.3)$$

$$\beta_{Q_2} = \frac{p_2 - x}{p_2 - p_1} \beta_2 + \frac{x - p_1}{p_2 - p_1} \beta_3 \quad (4.4)$$

そして、注視点での誤差を 2 点  $Q_1, Q_2$  での誤差から線形補間によって次のように求める。

$$\alpha = \frac{q_2 - y}{q_2 - q_1} \alpha_{Q_1} + \frac{y - q_1}{q_2 - q_1} \alpha_{Q_2} \quad (4.5)$$

$$\beta = \frac{q_2 - y}{q_2 - q_1} \beta_{Q_1} + \frac{y - q_1}{q_2 - q_1} \beta_{Q_2} \quad (4.6)$$

注視点の周りに基準点が存在しない場合には、注視点に最も近い基準点 4 点を探し、先ほどと同様にバイリニア補間によって誤差を求める。これによって得られた誤差だけ注視点をずらし

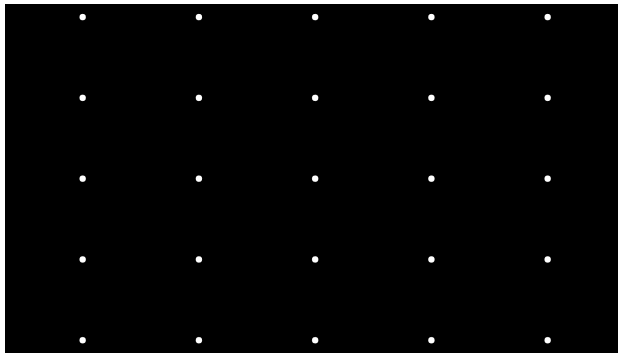


図 4.12 注視点データを取得する際にディスプレイに表示する基準点の位置関係

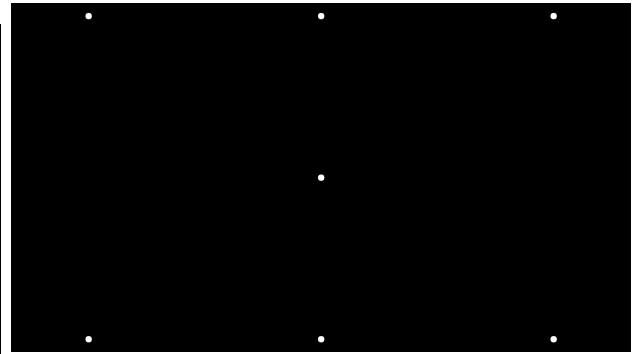


図 4.13 キャリブレーション結果を取得する際にディスプレイに表示する基準点の位置関係

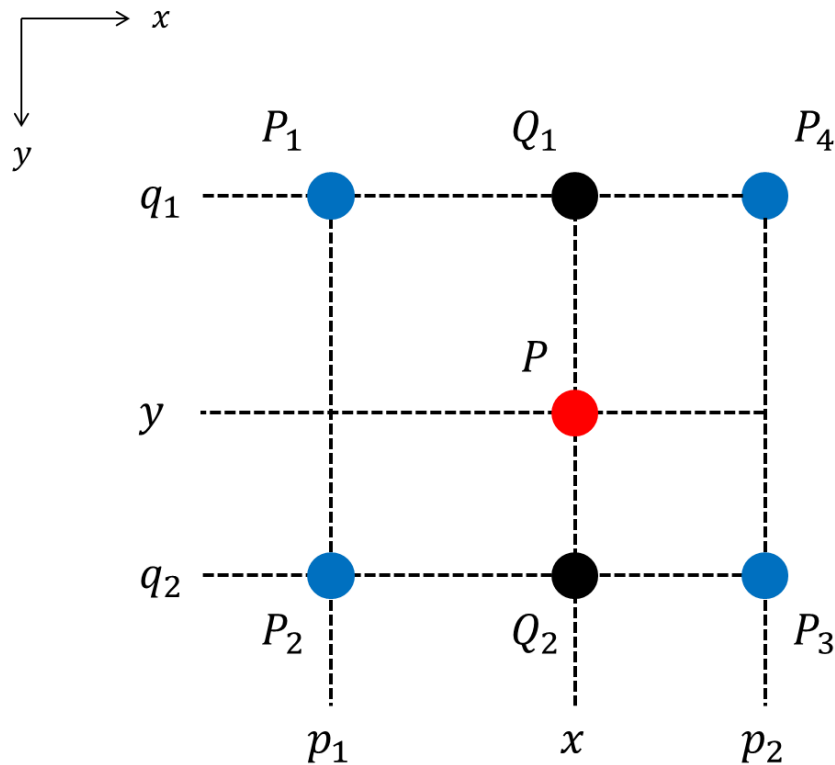


図 4.14 注視点とその周りの基準点の位置関係の例. 点  $P$  が注視点, 点  $P_i$  が基準点の位置を表している. また, 点  $Q_1$  は注視点  $P$  から直線  $P_1P_4$  に対して下した垂線の足, 点  $Q_2$  は注視点  $P$  から直線  $P_2P_3$  に対して下した垂線の足を表している.

た点  $(x + \alpha, y + \beta)$  を他ユーザのキャリブレーション結果を基に推定した注視点とし, これを未校正の注視点データとする.

ユーザの注視点データを取得する際の実験環境を図 4.15 に示す. ディスプレイは大きさ 23 インチ, 解像度  $1920 \times 1080$  のものを使用し, ユーザとディスプレイ間の距離は 60cm とし, ユーザの頭部はチンレストを用いて固定した. また, 注視点の推定には Tobii EyeX を使用し



図 4.15 ユーザの注視点データを取得する際の実験環境。ディスプレイは大きさ 23 インチ、解像度 1920×1080 のものを使用した。視線推定には Tobii EyeX を使用した。また、映像を閲覧する際にはユーザの頭部をチンレストを用いて固定し、ユーザとディスプレイ間の距離は 60cm とした。

た。ユーザのキャリブレーション結果は、ディスプレイ上に図 4.13 のような 7 点の基準点を順に表示し、それをユーザに注視させることによって取得した。そして、映像はなるべくディスプレイ全体に表示させるため、映像のスケール比は固定したまま拡大して表示し、得られた注視点データは元の映像の大きさに合うように縮尺した。このようにして得られた未校正の注視点データの例を図 4.16 に示す。

## 4.2.2 実験設定

4.1 節と同様に ASCMN データセットから 14 種類の映像を用いて、キャリブレーションの精度評価を行った。また、実験に際して 9 人のユーザの注視点データを収集した。入力 は 4.2.1 項で得られた未校正の注視点データとし、精度評価は、推定した変換パラメタを基に入力の注視点データを変換したものと、本人のキャリブレーション結果を基に推定した注視点



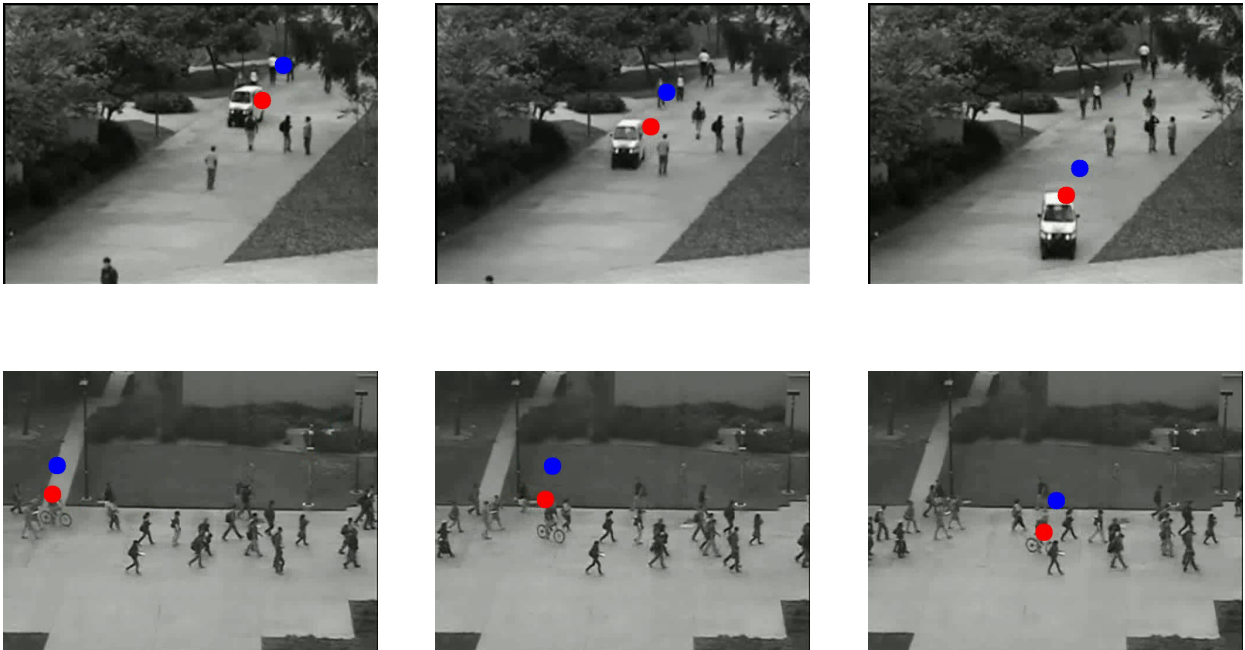


図 4.16 未校正の注視点データの例．赤点が真の注視点，青点が未校正の注視点を表している．上段では自動車，下段では自転車を注視しているが，未校正の注視点はそこからずれて位置している．

データとの RMSE を各映像のフレームサイズに対する割合に換算した値に基づいて行った．また，比較手法や実装については 4.1 節と同じである．

### 4.2.3 実験結果と考察

まず，図 4.17 に各映像での提案手法を用いたキャリブレーション結果を示す．図 4.17 から分かるように，擬似データでの結果と同様に，映像 S1 や S6 のときにはキャリブレーションの精度が良く，映像 L3 や L4 のときにはキャリブレーションの精度が悪い．

図 4.18 に提案手法と比較手法 1 との比較結果を示す．擬似データでの実験と同様に，比較手法 1 での結果は，複数のスケールで実験を行った際に精度が最も良かったスケールと最も悪かったスケールをデータごとに手動で選択した結果を示している．図 4.18 から擬似データでの結果とほぼ同様の結果が得られていることが分かる．また，比較手法 1 において選択されたスケールが視線の階層構造において何層目にあたるのか（図 4.19）を見ても，擬似データでの結果と同様にユーザごとに大きなばらつきがあることが確認できる．

次に，図 4.20 に提案手法と比較手法 2 との比較結果を示す．実際に収集した未校正の注視点データを入力としても，提案手法の方が水平方向，垂直方向ともに精度が良いことが分かり，顕著性を用いることの有効性が確認できる．一方，入力の未校正の注視点データと真の注視点データとの誤差とを比較すると，提案手法によるキャリブレーションを行うことで逆に

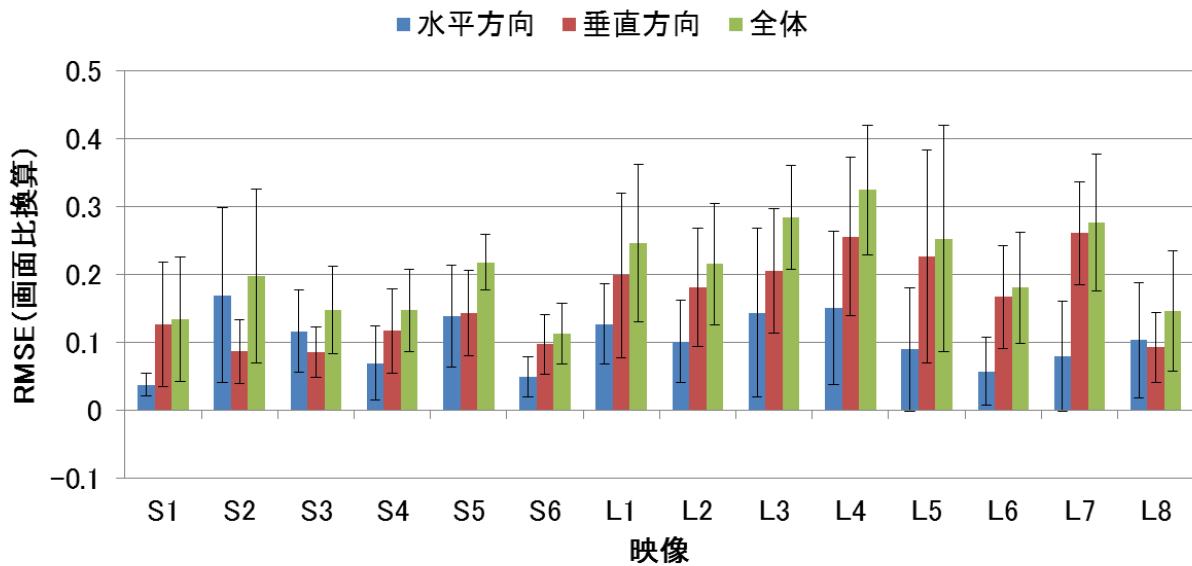


図 4.17 各映像での提案手法を用いたキャリブレーション結果. S1 から S6 が短い映像, L1 から L8 が長い映像を表している.

誤差が大きくなっていることが分かる. これは 4.1.4 節でも述べたように, 注視しているオブジェクト全体が同じフローで動いている場合にはそのオブジェクト内のどこを注視しているかまでは提案手法では特定できないことが原因の一つであると考えられる. また, 実際には滑動性眼球運動を行っていないのに誤って検出し, その区間から真の注視領域とは全く異なる領域を注視領域として推定しているということも考えられる.

擬似データでの結果と同様な結果が得られていることから, 未校正の注視点データは真の注視点データに対してスケールとバイアスが水平方向と垂直方向に対してそれぞれ独立にかかっているものである, という仮定は妥当であるといえる. また, 提案手法はスケールとバイアスの大きさに依存しにくい手法であると考えられ, スケールとバイアスがさらに大きくなった場合でも適用できるといえる.

また, 図 4.21 に具体的なキャリブレーション結果の一例を示す. 図 4.21 の上段の映像は, 人が右方向から左方向へ歩行する映像であり, 赤枠内のようにその領域を注視領域として推定することができていることが分かる. 一方, 図 4.21 の下段の映像は, 複数人が歩行する映像であるが, ユーザは赤枠内の左側の人物を注視しているのに対し, 推定した注視領域は右側の人物であることが水平方向での結果から見て取れる. これは, どちらの人物も似た動きをしており, 動きの顕著性も高いため注視領域を一意に推定できなかったことが原因であると考えられる.

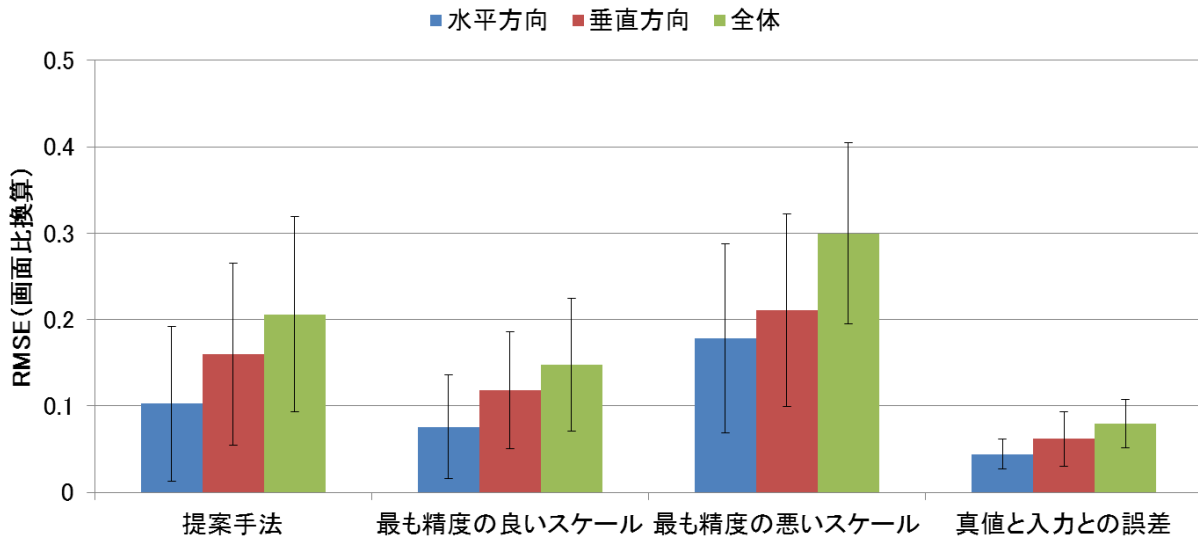


図 4.18 提案手法と比較手法 1 との比較結果. 比較手法 1 では, 複数のスケールで実験を行い, その中で最も精度の良かったスケールと最も精度の悪かったスケールをデータごとにそれぞれ手動で選択した際の結果を表している. また, 真の注視点データと入力の注視点データとの誤差も示している.

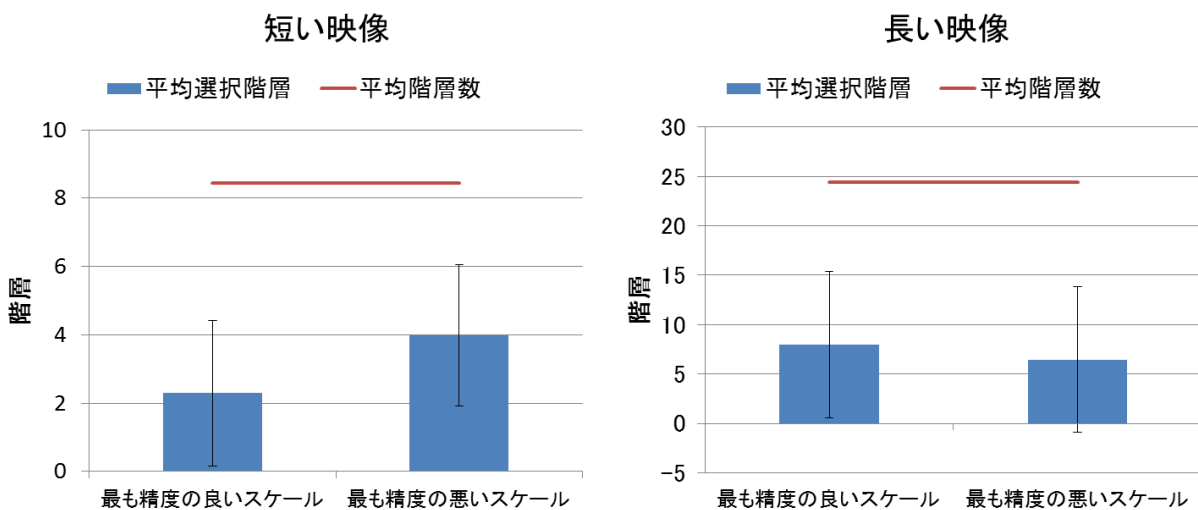


図 4.19 比較手法 1 において選ばれたスケールに相当する視線の階層構造における階層. 左が短い映像での結果, 右が長い映像での結果を表している.

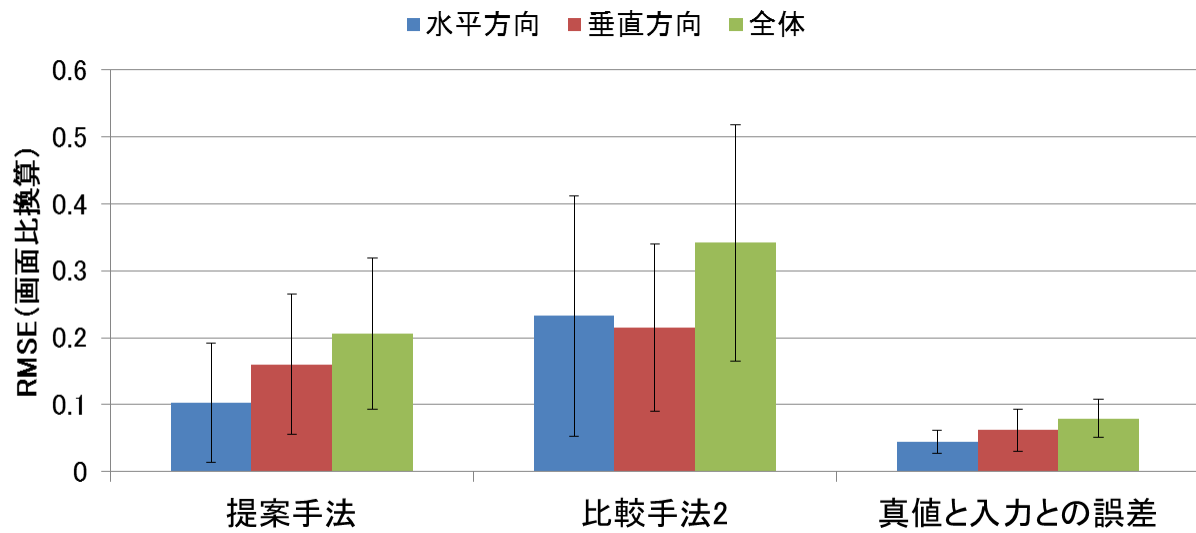


図 4.20 提案手法と比較手法 2 との比較結果. また, 真の注視点データと入力の注視点データとの誤差も示している.

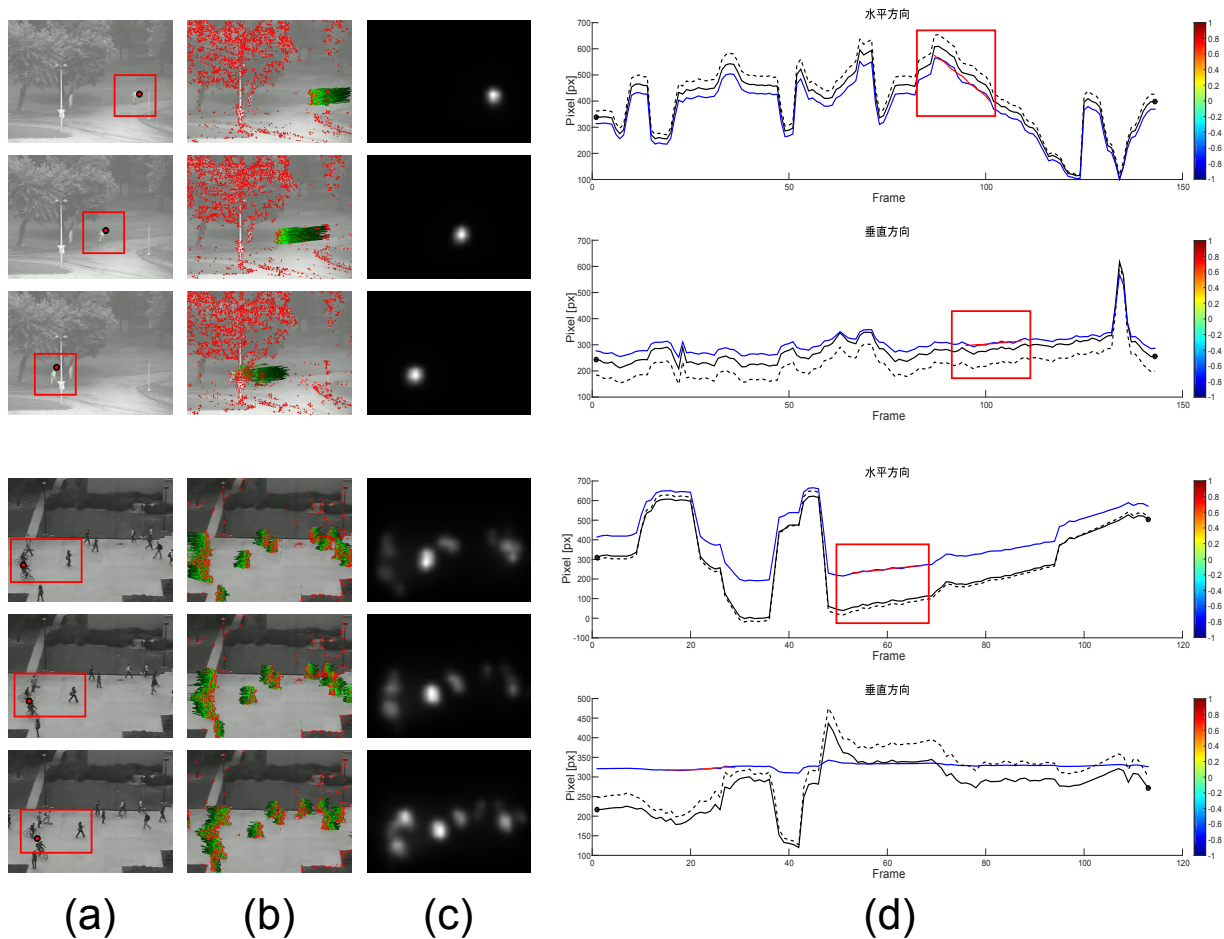


図 4.21 ユーザごとのキャリブレーション結果. (a) 映像の数フレーム. フレーム内の赤点は真の注視点を表している. (b)Dense trajectory. (c) 顕著性マップ. (d) キャリブレーション結果. グラフは上段が水平方向での結果, 下段が垂直方向での結果を表している. また, 黒線が真の注視点データ, 青線は推定した変換パラメタを基に推定した注視点データ, 黒の点線は入力注視点データ, 色のついた短線はキャリブレーションに用いた注視領域を表している. グラフ内での注視領域の色は評価値の大きさに基づいており, 評価値が高いほど赤く, 低いほど青くなる. そして, 黒点に囲まれた時区間が整数計画問題を解くことによって選ばれた分節を表している.

## 第5章

# 結論

### 5.1 まとめ

本研究では、視線計測におけるキャリブレーションの自動化を目的として、ユーザが一般的な映像を閲覧する際にキャリブレーションを行う手法を提案した。提案手法では、滑動性眼球運動に着目し、眼球運動と映像中の動きとの相関および映像中の動きの顕著性から注視領域を推定することでキャリブレーションを行った。また、擬似的に生成した未校正の注視点データを入力としたキャリブレーションの精度評価を行い、提案手法の有効性を検証した。そして、実際にユーザが映像を閲覧する際の未校正の注視点データを収集しキャリブレーションの精度評価を行うことで、実データに対してもその有効性を検証した。

### 5.2 課題と今後の展望

#### 5.2.1 未校正の注視点データの定義

本研究では、真の注視点データに対してスケールとバイアスが水平方向と垂直方向に対してそれぞれ独立にかかっているような注視点データを未校正の注視点データとして想定した。しかし、現実的には回転変換などのようにスケールとバイアスのみでは表現できないような変換がかけられている場合も想定される。例えば、図 5.1 は、あるユーザがディスプレイ上の基準点を注視する際に得られたデータから生成した、未校正の注視点と真の注視点とのエラーマップを表している。これを見ると、誤差の変化は不規則であり、単純なスケールとバイアスのみでは表せないことが分かる。

本研究では映像中の動きと眼球運動との相関として正規化相互相関を用いているが、例えば回転変換がかけられている場合には正規化相互相関では真の注視点の動きとは全く方向の異なる領域の相関が高くなるという問題が生じる。そのため、この場合には速度の変化量を基に相関を計算するといったように相関の計算手法を変更する必要がある。ただし、このとき、水平

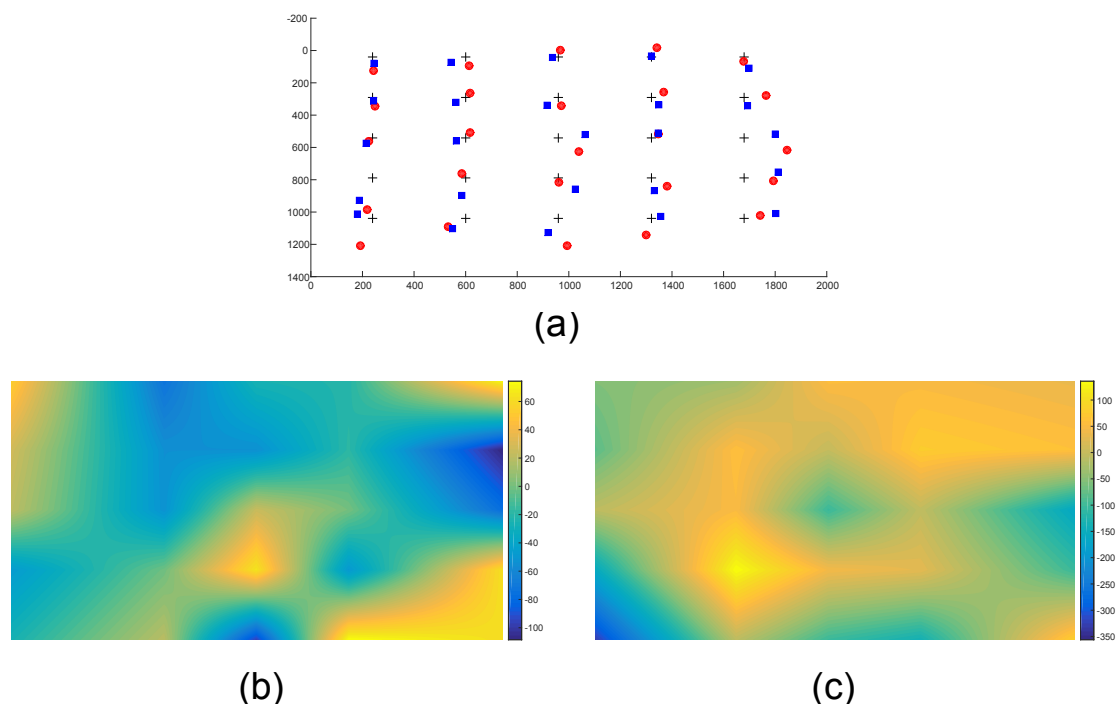


図 5.1 未校正の注視点データと真の注視点データとのエラーマップ。(a) はディスプレイ上の基準点 (+) をユーザが注視しているときの推定した注視点を表している。赤点は真の注視点、青点は未校正の注視点を示す。この各基準点での注視点の誤差を基に補間によってディスプレイ全体へと拡張したものが (b) と (c) である。(b) は水平方向の誤差、(c) は垂直方向の誤差を表している。未校正の注視点データが真の注視点データに対してスケールとバイアスのみの変換によって得られるのであれば、このエラーマップはある 1 点から同心円状に色が変わることになる。

方向の動きと垂直方向の動きを区別できなくなるため更なる工夫が必要であると考えられる。また、図 5.1 のように非線形変換がかけられている場合にも対処できるような手法も考える必要がある。

## 5.2.2 注視領域候補の選択方法

提案手法では、キャリブレーションに適した滑動性眼球運動候補と注視領域候補を選ぶ際に、整数計画問題の定式化上、1 つの滑動性眼球運動候補に対して必ず 1 つの注視領域候補を選ぶ必要がある。そのため、細かく分節化された滑動性眼球運動候補の一部に対して評価値の高い注視領域候補がない場合、複数の滑動性眼球運動候補と注視領域候補を選ぶよりも区間全体を滑動性眼球運動として注視領域候補を 1 つ選んだ方が全体で見たときに評価値の和が大きくなる。しかし、この選ばれた 1 つの注視領域候補が真の注視点に対して全く位置の異なるものであった場合、図 4.11 の下段の垂直方向での結果のようにキャリブレーションに失敗す

るといった状況が生じる。したがって、1つの滑動性眼球運動候補に対して必ず1つの注視領域候補を選ばなくてもよいようにする必要がある。これによって、複数の注視領域候補が選ばれ、その中の1つが真の注視点に対して位置が全く異なっていたとしても、他の注視領域候補によってその失敗を補うことができると考えられる。

また、複数の注視領域候補を選ぶようにする手段として、視線の階層構造の各分節に加える重みを調節する手段が考えられる。本研究では、各分節が自身の下の層で最終的に分節化される個数をその分節の重みとしたが、この重みが大きすぎると区間全体を滑動性眼球運動として選択される場合が多くなる。そのため、この重みについては更なる検討が必要である。

一方、整数計画問題において二次の項を加える手段も考えられる。本研究では、注視領域候補の評価値の和を一次の項とし、二次の項を加えないような整数計画問題を解くことによって注視領域を推定した。キャリブレーションを行う際には、ディスプレイの広範囲に渡って基準点と眼球情報との対応関係が得られた方がより高精度の視線推定を行うことができる。これを考慮すると、例えば整数計画問題における二次の項として選ばれる注視領域候補同士の距離が離れているほど値が大きくなるような項を加えることによって、評価値が高く候補同士がなるべく離れているような注視領域を推定することができる。

### 5.2.3 オンライン処理によるキャリブレーション

本研究では、映像を閲覧する際の未校正の注視点データ全てを獲得した後に、それを入力としたオフライン処理によってキャリブレーションを行った。しかし、注視点推定を行う際には、ユーザの頭部の位置などがキャリブレーション時に比べ時間とともに変化するため、推定する注視点に誤差が蓄積する。この問題を解消する一つのアプローチとして、オンライン型のキャリブレーションが考えられる。すなわち、ユーザが映像を閲覧する際において一定区間ごとに眼球運動との相関が高く、かつ顕著性も高いような領域を注視領域として推定することで、キャリブレーション結果をオンラインで更新できる。このような技術が確立されれば、例えばテレビ番組を閲覧する際のユーザのキャリブレーションをコマーシャル映像中に行うといったシステムが実現でき、視線情報に基づく番組への興味推定や集中度推定など、いくつかの重要な応用シーンに有効利用できると見込まれる。



## 参考文献

- [1] F. Alnajar, T. Gevers, R. Valenti, and S. Ghebreab. Calibration-Free Gaze Estimation Using Human Gaze Patterns. In *Proceedings of IEEE International Conference on Computer Vision*, pp. 137–144, 2013.
- [2] S. Baluja and D. Pomerleau. Non-Intrusive Gaze Tracking Using Artificial Neural Networks. *Advances in Neural Information Processing Systems*, Vol. 6, pp. 753–760, 1994.
- [3] J.J. Cerrolaza and A. Villanueva. Taxonomic Study of Polynomial Regressions Applied to the Calibration of Video-oculographic Systems. In *Proceedings of Eye Tracking Research and Applications*, pp. 259–266, 2008.
- [4] J.J. Cerrolaza, A. Villanueva, M. Villanueva, and R. Cabeza. Error Characterization and Compensation in Eye Tracking Systems. In *Proceedings of Eye Tracking Research and Applications*, pp. 205–208, 2012.
- [5] J. Chen and Q. Ji. Probabilistic Gaze Estimation Without Active Personal Calibration. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 609–616, 2011.
- [6] J. Choi, B. Ahn, J. Park, and I. Kweon. Gmm-based saliency aggregation for calibration-free gaze estimation. In *Proceedings of IEEE International Conference on Image Processing*, pp. 1096–1099, 2014.
- [7] A. Esteves, E. Velloso, A. Bulling, and H. Gellersen. Orbits: Gaze Interaction for Smart Watches using Smooth Pursuit Eye Movements. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software and Technology*, pp. 457–466, 2015.
- [8] G. Farnebäck. Two-Frame Motion Estimation Based on Polynomial Expansion. In *Proceedings of the 13th Scandinavian Conference on Image Analysis*, pp. 363–370, 2003.
- [9] E.D. Guestrin and M. Eizenman. General Theory of Remote Gaze Estimation Us-

- ing the Pupil Center and Corneal Reflections. *IEEE Transactions on Biomedical Engineering*, Vol. 53, No. 6, pp. 1124–1133, 2006.
- [10] D.W. Hansen and Q. Ji. In the Eye of the Beholder: A Survey of Models for Eyes and Gaze. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 32, No. 3, pp. 478–500, 2010.
- [11] J. Harel, C. Koch, and P. Perona. Graph-based visual saliency. In *Proceedings of Advances in Neural Information Processing Systems*, Vol. 19, pp. 545–552, 2006.
- [12] C. Hennessey, B. Nouredin, and P. Lawrence. A Single Camera Eye-Gaze Tracking System with Free Head Motion. In *Proceedings of Eye Tracking Research and Applications*, pp. 87–94, 2006.
- [13] L. Itti, C. Koch, and E. Niebur. A Model of Saliency-Based Visual Attention for Rapid Scene Analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20, No. 11, pp. 1254–1259, 1998.
- [14] R.J.K. Jacob. The Use of Eye Movements in Human-computer Interaction Techniques: What You Look at is What You Get. *ACM Transactions on Information Systems*, Vol. 9, No. 3, pp. 152–169, 1991.
- [15] Q. Ji and X. Yang. Real-Time Eye, Gaze, and Face Pose Tracking for Monitoring Driver Vigilance. *Real-Time Imaging*, Vol. 8, No. 5, pp. 357–377, 2002.
- [16] O. Komogortsev and A. Karpov. Automated classification and scoring of smooth pursuit eye movements in the presence of fixations and saccades. *Behavior Research Methods*, Vol. 45, No. 1, pp. 203–215, 2013.
- [17] L. Larsson, M. Nyström, R. Andersson, and M. Stridh. Detection of fixations and smooth pursuit movements in high-speed eye-tracking data. *Biomedical Signal Processing and Control*, Vol. 18, pp. 145–152, 2015.
- [18] D. Model and M. Eizenman. An Automatic Personal Calibration Procedure for Advanced Gaze Estimation Systems. *IEEE Transactions on Biomedical Engineering*, Vol. 57, No. 5, pp. 1031–1039, 2010.
- [19] C.H. Morimoto and M.R.M. Minica. Eye Gaze Tracking Techniques for Interactive Applications. *Computer Vision and Image Understanding*, Vol. 98, No. 1, pp. 4–24, 2005.
- [20] T. Ohno. One-point Calibration Gaze Tracking Method. In *Proceedings of Eye Tracking Research and Applications*, pp. 34–34, 2006.
- [21] K. Pfeuffer, M. Vidal, J. Turner, A. Bulling, and H. Gellersen. Pursuit Calibration: Making Gaze Calibration Less Tedious and More Flexible. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*, pp. 261–270,

- 2013.
- [22] K. Rayner, C.M. Rotello, and A.J. Stewart. Integrating Text and Pictorial Information: Eye Movements When Looking at Print Advertisements. *Journal of Experimental Psychology*, Vol. 7, No. 3, pp. 219–226, 2001.
- [23] N. Riche, M. Mancas, D. Culibrk, V. Crnojevic, B. Gosselin, and T. Dutoit. Dynamic saliency models and human attention: a comparative study on videos. In *Proceedings of the 11th Asian Conference on Computer Vision*, No. 586–598, 2012.
- [24] J. Shi and C. Tomasi. Good features to track. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 593–600, 1994.
- [25] A.M. Slater and J.M. Findlay. The Measurement of Fixation Position in the Newborn Baby. *Journal of Experimental Child Psychology*, Vol. 14, pp. 349–364, 1972.
- [26] Y. Sugano, Y. Matsushita, and Y. Sato. Calibration-free gaze sensing using saliency maps. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2667–2674, 2010.
- [27] Y. Sugano, Y. Matsushita, and Y. Sato. Appearance-Based Gaze Estimation Using Visual Saliency. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 35, No. 2, pp. 329–341, 2013.
- [28] Y. Sugano, Y. Matsushita, Y. Sato, and H. Koike. An Incremental Learning Method for Unconstrained Gaze Estimation. In *Proceedings of the 10th European Conference on Computer Vision*, pp. 656–667, 2008.
- [29] K.-H. Tan, D.J. Kriegman, and N. Ahuja. Appearance-based Eye Gaze Estimation. In *Proceedings of the Sixth IEEE Workshop on Applications of Computer Vision*, pp. 191–195, 2002.
- [30] M. Vidal, A. Bulling, and H. Gellersen. Pursuits: Spontaneous Interaction with Displays based on Smooth Pursuit Eye Movement and Moving Targets. In *Proc. of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pp. 439–448, 2013.
- [31] M. Vidal, K. Pfeuffer, A. Bulling, and H. Gellersen. Pursuits: Eye-based Interaction with Moving Targets. In *Ext. Abstr. of the 31st SIGCHI International Conference on Human Factors in Computing Systems*, pp. 3147–3150, 2013.
- [32] H. Wang, A. Klaser, C. Schmid, and C.-L. Liu. Action Recognition by Dense Trajectories. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3169–3176, 2011.
- [33] O. Williams, A. Blake, and R. Cipolla. Sparse and Semi-supervised Visual Mapping with the  $S^3GP$ . In *Proceedings of IEEE Conference on Computer Vision and Pattern*

- Recognition*, Vol. 1, pp. 230–237, 2006.
- [34] A. Witkin. Scale-space filtering. In *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, No. 1019–1022, 1983.
- [35] L.-Q. Xu, D. Machin, and P. Sheppard. A Novel Approach to Real-time Non-intrusive Gaze Finding. In *Proceedings of British Machine Vision Conference*, 1998.
- [36] R. Yonetani, H. Kawashima, T. Hirayama, and T. Matsuyama. Gaze Probing: Event-Based Estimation of Objects Being Focused On. In *Proceedings of IEEE International Conference on Pattern Recognition*, pp. 101–104, 2010.
- [37] 米谷竜, 川嶋宏彰, 加藤丈和, 松山隆司. 映像の顕著性変動モデルを用いた視聴者の集中状態推定. 画像の認識・理解シンポジウム (MIRU2012) , 2012.

# 発表文献

## 国内会議（査読なし）

- [1] 松本大輝, 米谷竜, 佐藤洋一, “滑動性眼球運動を用いた視線計測の自動校正,” 画像の認識・理解シンポジウム (MIRU2015), 2015.
- [2] 松本大輝, 米谷竜, 佐藤洋一, “滑動性眼球運動を用いた視線計測の自動校正,” コンピュータビジョンとイメージメディア研究会 (CVIM200), 2016.

## 本研究に含まれない文献

### 国際会議

- [3] D. Matsumoto, Y. Matsui, T. Yamasaki, K. Aizawa and T. Katagiri. *IllustStyleMap: Visualization of Illustrations Based on Similarity of Drawing Style of Authors*. In *ACM SIGGRAPH 2014 Posters*, 2014.

## 国内会議（査読なし）

- [4] 松本大輝, 松井勇佑, 山崎俊彦, 相澤清晴, 片桐孝憲, “著者の画風の類似性に基づくイラスト画像の可視化,” パターン認識とメディア理解研究会 (PRMU), 2014.

# 謝辞

本研究を進めるにあたり，ご多忙の中，定期的にミーティングの機会を設けていただき，多大なる御指導を賜りました東京大学生産技術研究所の佐藤洋一教授に深く感謝いたします。

東京大学生産技術研究所の米谷竜助教には，研究方針を立てる上で終始的確なご助言を頂いただけでなく，研究会発表や本論文の執筆に際しても懇切丁寧なご指導を頂きました。本当にありがとうございました。

秘書の鈴木咲恵さん，今川洋子さんには様々な事務手続きなどでお世話になり，何不自由なく研究生生活を送ることができました。深くお礼を申し上げます。

また，本研究で使用した実験データの収集にあたり，佐藤研究室のメンバーの方々にご協力頂きました。本当にありがとうございました。

そして，佐藤研究室の皆様，特に同期としてお互いに助け合った，神窪利絵さん，杉田祐樹くん，村上晋太郎くん，Nattawan Tantirujananont さん，2年間ありがとうございました。

最後に，ずっと暖かく支えてくださった両親に感謝します。

2016年2月3日