

修士論文

広視野な車載単眼カメラを用いた 歩行者追跡に関する研究

A Study on Pedestrian Tracking by On-Board Wideview Monocular camera

指導教官 上條 俊介 准教授



東京大学大学院
情報理工学系研究科
電子情報学専攻

学籍番号・氏名 096410 柴山 悠毅

提出日 平成 23 年 2 月 9 日

目次

第1章 序論.....	1
1.1 研究背景.....	2
1.2 研究目的.....	2
1.3 本論文の構成.....	5
第2章 関連研究.....	6
2.1 路者協調型事故防止技術.....	7
2.2 車載センサによる事故防止技術.....	7
2.2.1 外見情報を用いた歩行者検出技術の研究例.....	8
2.2.2 動き情報による歩行者検出技術.....	8
2.2.3 近年の歩行者検出技術の研究.....	9
2.3 本研究の位置付け.....	9
第3章 時空間 MRF モデルによるオブジェクト検出・トラッキング.....	11
第4章 オブジェクト判別.....	20
4.1 幾何拘束によるオブジェクト選別.....	21
4.1.1 オブジェクト ROI の高さ推定.....	21
4.1.2 ROI 高さ”H”の推定.....	27
4.2 HOG カスケード判別器.....	30
4.2.1 HOG 特徴量について.....	30
4.2.2 学習データ構成.....	31
4.2.3 判別器の性能検証について.....	36
4.3 セマンティック HOG 特徴量による更なる選別.....	38
第5章 歩行者の車両に対する相対軌跡の導出.....	43
第6章 ROI 情報修正アルゴリズム.....	52
6.1 オブジェクトの ROI 情報の修正.....	53
6.1.1 ROI 拡張アルゴリズム.....	54
6.1.2 ROI 分割アルゴリズム.....	56
6.2 ROI 修正アルゴリズム適用結果.....	58
第7章 実験結果と今後の課題.....	69
7.1 実験結果.....	70
7.1.1 対象シーンについて.....	70
7.1.2 評価手法および性能評価.....	70
7.2 結果考察.....	81
7.3 まとめと今後の課題.....	82
謝辞.....	84

参考文献.....	85
参考文献.....	86

図目次

図 1.1	交通事故発生件数, 及び死傷者数の推移(平成 22 年度).....	2
図 1.2	想定シーン.....	4
図 1.3	カメラ設置図.....	4
図 3.1	動きベクトル初期エラー状態と最適化状態.....	14
図 3.2	ブロックライン上での動きベクトル最適化.....	15
図 3.3	時空間 MRF モデルによるオブジェクト検出結果.....	19
図 4.1	シーン撮影時のカメラ設置状況.....	22
図 4.2	1m 当たりのピクセル数 h_p [pixels / meter] と カメラからの距離 D [meters].....	23
図 4.3	位置(x, y) [pixels]とカメラからの距離 D [meters].....	26
図 4.4	収集されたカメラからの距離 D [meter]に存在する人の身長 H [meter].....	29
図 4.5	HOG 特徴量をヒストグラム化した画像.....	31
図 4.6	学習データ構成.....	35
図 4.7	セマンティック HOG 特徴量について.....	38
図 4.8	元画像, HOG 特徴量及びセマンティック HOG 特徴量.....	39
図 4.9	セマンティック HOG 特徴量により歩行者・非歩行者オブジェクト間の差.....	40
図 4.10	セマンティック HOG 非線形判別器の特徴探索例.....	41
図 5.1	相対軌跡検証実験 直進シーン.....	45
図 5.2	相対軌跡検証実験 右折シーン.....	46
図 5.3	正解データ(赤線)・絶対軌跡(赤点)・相対軌跡(青点).....	47
図 5.4.1	相対軌跡取得例 ケース 1	49
図 5.4.2	相対軌跡取得例 ケース 2	51
図 6.1	ROI 情報拡張アルゴリズム.....	54
図 6.2	ROI 情報分割アルゴリズム.....	56
図 6.3.1	ROI 拡張アルゴリズム適用例 ケース 1	59
図 6.3.2	ROI 拡張アルゴリズム適用例 ケース 2	60
図 6.3.3	ROI 拡張アルゴリズム適用例 ケース 3	62
図 6.3.4	ROI 拡張アルゴリズム適用例 ケース 4	64
図 6.4.1	ROI 分割アルゴリズム適用例 ケース 1	65
図 6.4.2	ROI 分割アルゴリズム適用例 ケース 2.....	66
図 6.4.3	ROI 分割アルゴリズム適用例 ケース 3.....	67
図 6.4.4	ROI 分割アルゴリズム適用例 ケース 4.....	68

図 7.1.1	実験結果画像	シーン 1 : 左折	左カメラ	75°	レンズ使用.....	74
図 7.1.2	実験結果画像	シーン 2 : 直進	右カメラ	60°	レンズ使用.....	76
図 7.1.3	実験結果画像	シーン 3 : 左折	左カメラ	75°	レンズ使用.....	78
図 7.1.4	実験結果画像	シーン 4 : 右折	右カメラ	30°	レンズ使用.....	80

表目次

表 4.1	SVM と Fisher の判別性能比較結果.....	37
表 7.1	アルゴリズム性能評価.....	72

第 1 章 序論

本研究では、これまで本研究室が開発してきた時空間 MRF モデルによるオブジェクトトラッキングをベースに、広視野な単眼車載カメラを用いた道路上の歩行者の検出・追跡技術の研究を行った。

本章では、本研究の背景と目的、本論文の構成について述べる。

1.1 研究背景

近年，シートベルト着用の徹底・義務化や飲酒運転等の悪質な運転の取締りの強化，エアバッグなどの衝突時安全装置や ABS (Antilock Brake System) などの予防安全装置の普及，また医療技術の向上による救命装置の高度化などにより，交通事故における死亡事故・重大事故の割合は減少傾向にあるといえる．事実，最新の統計[1]では交通事故件数・負傷者数では 7 年連続の減少傾向にあり，交通事故死者数は 57 年ぶりに 5000 人を下回っている．しかし，減少傾向にあるといえど交通事故は未だ重大な社会問題であるといえる．なぜならば，突発的に起こり得る交通事故は加害者・被害者双方に与える精神的・肉体的被害が甚大な事や，家屋や構造物への衝突による破損など直接的な被害，また事故渋滞による間接的な経済的損失も大きく，平成 11 年度の統計によると国内の交通事故渋滞による経済的損失額は 1500 億円に上るとされており，交通事故件数を出来る限り 0 に近づける事は，未だ社会が抱える重大な社会的な課題といえるからである．

医療，環境，災害などの諸問題と共に交通分野での社会的課題を解決すべく 2006 年 1 月の政府 IT 新改革戦略[2]が発案・実施された．これを受けて ITS (Intelligent Transport Systems) [3]技術を活用し，交通事故減少を目標に，AHS など[4]の様々な路車間協調の社会実験が行われた．しかし路車間協調のシステムは交通量の多く事故が多発している交差点や道路などでは高い効果が期待できるものの，路側（インフラ側）にシステムが実装されているエリアでのみしか効果を発揮できない，実装のコストやシステム自体のメンテナンス費なども高い，などの問題点が挙げられる．以上の事由から日本のみならず，欧米でも車載センサによる歩行者検出技術の需要が高まっている．

車載センサによる歩行者検出技術としてはすでに様々な研究がなされている．用いるセンサの種類としては超音波センサやレーザーレーダー，そして画像センサなどが用いられることが一般的である．しかし超音波センサは空間分解能が低く，レーザーレーダーはレーダー前を横切らないと感知しない，などといった問題点があり，これらを単体で使用するには効果が不十分である．一方で、画像センサを用いた手法では上記の問題をクリアしており，他センサより優位な状況にある．

この画像センサによる認識技術の有用性に着目した研究が，従来から数多く行われてきた．しかし，コストの高いステレオカメラシステムを採用していることや，主対象とするシーンが前方の飛び出しを想定しているシステムがほとんどであった．特に，日本など交差点での巻き込み事故が対歩行者事故数の 60%を占めるような状況での実運用に耐えるシステムは実用化されていない．理由としては画像センサを用いた場合、背景が単純な直線走行時での歩行者検出はまだしも，交差点上のように撮影背景が任意に動きまた非常に複

雑である場合には、歩行者領域の検出が難しいといった問題が存在するためである。そこで本研究では、移動物体のトラッキング技術として、これまで上條研究室が開発してきた時空間 MRF 技術を基礎として用いている。時空間 MRF 技術とは、画像を時系列に蓄積した時空間画像において錯綜して重なり合う複数の移動物体の領域を分割する確率モデルである。当該技術によれば、画像上で重なり合う（オクルージョン）複数の物体を分離して個別に追跡することができるため、多数の移動物体が画像上で重なりながら往来する交差点上での運用も耐えうるシステムの開発が期待できる。

また、時空間MRF 技術では、既存のトラッキングアルゴリズムと異なり、車や人といった移動物体固有の形状やエッジといった特徴量を用いていないため、車や人等の全く異なる形状をした物体を単一アルゴリズムで扱うことが可能である。これにより、シンプルかつ拡張性のあるシステムを構築できるという利点がある。

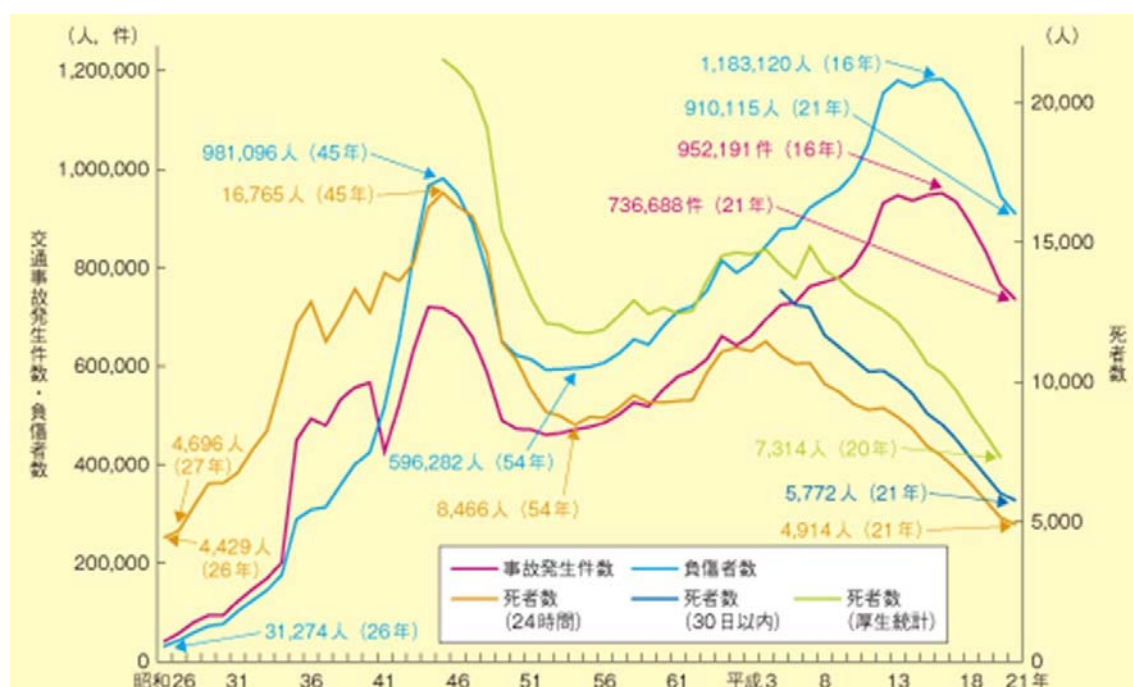


図1.1 交通事故発生件数、及び死傷者数の推移（平成22年度）

1.2 研究目的

車載画像センサを用いた道路上の歩行者検出技術の研究は広くなされている。しかしこれらの研究の多くは直線道路を走行中に、前方で飛び出してくる歩行者を主な検出対象とする研究がほとんどである。これは交差点での歩行者検出技術は直線道路での前方飛び出し検知技術より技術的に困難であることと、道路面積が狭く、整備がなされていない道路

の多い日本に比べ、広く見通しの良い道路が多く存在するヨーロッパやアメリカでは、前方飛び出し検出技術のニーズが高いことが原因として挙げられる。しかし交差点での交通事故割合が交通事故件数のうち 60%を占める日本では、歩行者検出の技術の必要性が欧米に比べ高いといえる。また、高精度の検出性能を達成するためにステレオカメラシステムや複数センサをフュージョンさせて用いている研究が多数存在するが、前方左右それぞれにマルチセンサシステムを搭載するのはコスト的に問題があるといえる。これらの事由から本研究では、図 1.2 に示すように前方監視に限らず交差点上での車両周辺も含めた歩行者検出・追跡を単眼カメラで行う事を目的にしている。また、その際のカメラ設置図を図 1.3 に示す。後半で示す本研究の性能評価は、全て図 1.2, 1.3 に示すような状況・カメラ設定で撮影されたシーンを用いている。

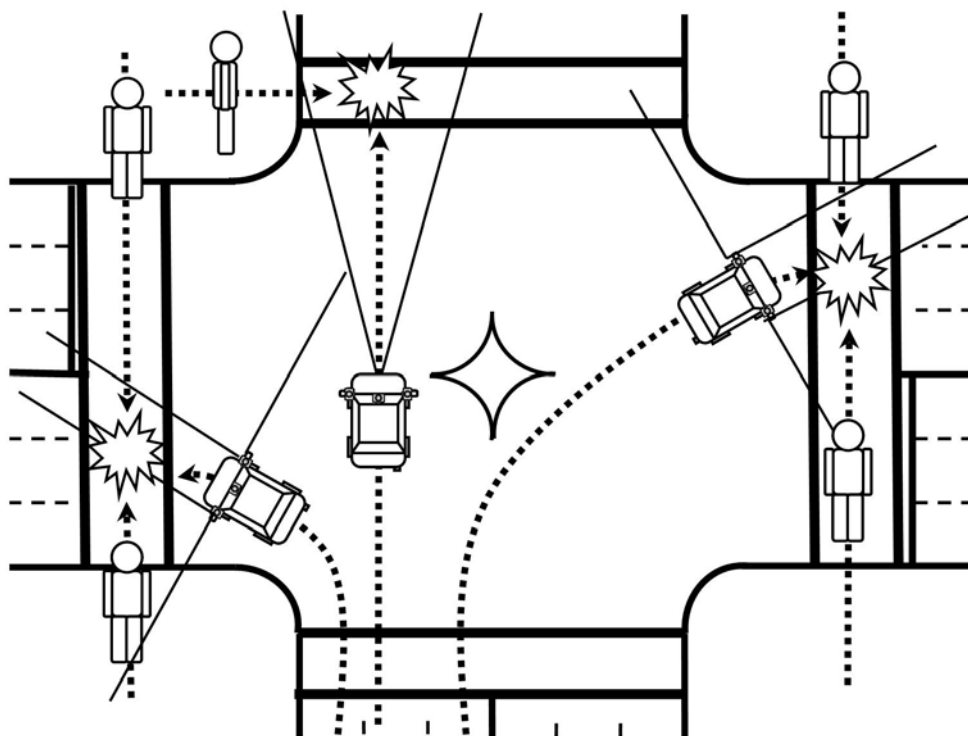


図 1.2 想定シーン

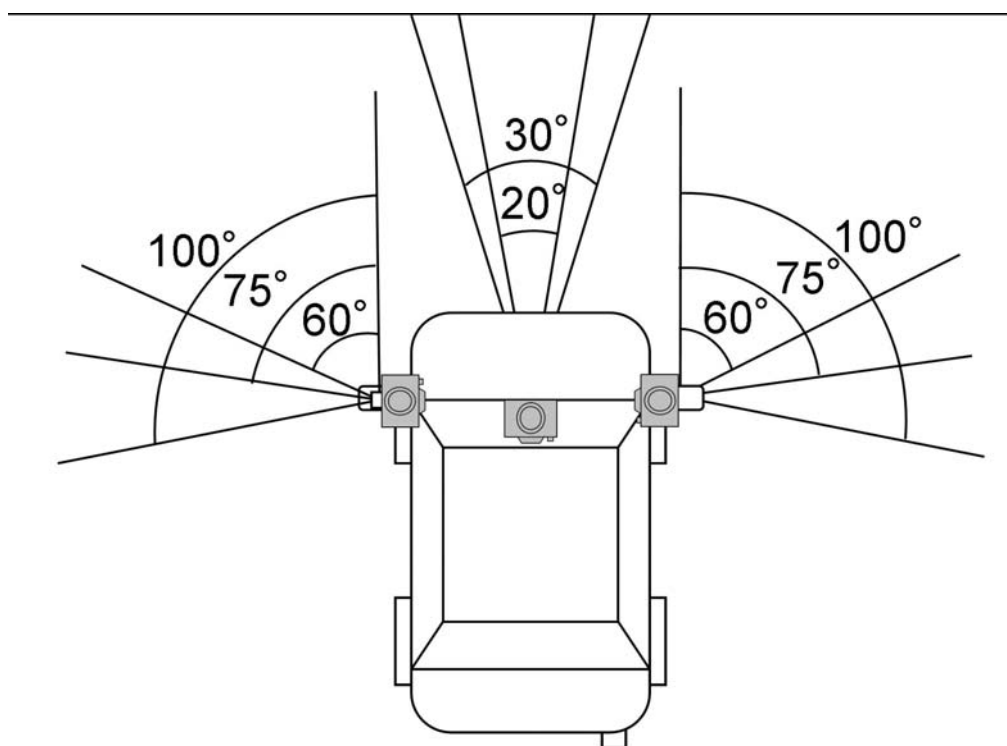


図 1.3 カメラ設置図

1.3 本論文の構成

本章では研究に至る背景とその目的，本論文の構成について述べた．

続く第2章では車載センサを用いた道路上の歩行者検出技術の従来に関連技術，研究を説明する．

第3章では本研究室で開発されてきた時空間 **MRF** モデルを用いての，オブジェクト検出，トラッキング技術について説明する．

第4章では検出されたオブジェクトを歩行者か否か判別する判別器の構成について説明する．

第5章では検出された歩行者の車両に対する相対軌跡を導出し，その正確性の検証を行った．

第6章では，歩行者候補領域情報に修正を加えることにより更なる結果の向上を試み，その結果を確認する．

そして最後に第7章で結論と，今後のまとめを述べる．

第2章 関連研究

交通事故を未然に防止する路者協調型技術の研究として，現在様々な社会実験が行われている．また世界各地では画像センサだけでなく，センサ自体を複数用いて欠点を補い合うなど様々な手法を用いて車載センサを用いた歩行者検出技術が研究されている．本章ではそれらの本研究に関連する研究について述べる．

2.1 路者協調型事故防止技術

道路上にセンサ類を設置し、センサからの情報を元に交通監視を行う路者強調型システムによる安全運転支援技術の研究は古くからなされている。路者協調型のシステムの代表的な例として、事象検出が挙げられる。事象検出の目的とは道路上で発生した渋滞、事故などの事象を後続の車両に伝え更なる渋滞、事故を防ぐことにある。事象検出を目的とした研究では、道路上に様々なセンサが配置できることから、現在センサ融合手法を用いた研究が一般的である。たとえば[5]のように実際の高速道路にセンサフュージョンシステムを実装し、その結果も実際のシステムとして運用している研究も存在する。またその他にも、道路上の車両から見えない横断中の歩行者などを検出し、車両に伝える技術の研究もされており、こちらも社会実験[6]を行っている。

路車協調型システムによる安全支援技術は、設置された地点において非常に高精度な結果が期待できる。しかし問題点として以下の点が挙げられる。第一に設置されたシステムをメンテナンスするコストが高い、という点である。現在システムを設置している地点は大規模な地点が多くなっているが、それに応じてメンテナンスコストも非常に高くなってしまふ。第二にシステムによる支援が、システムが実装された地点でのみしか受けられない点である。アメリカやヨーロッパなどの広く整備された道路状況とは異なり、日本では都心や国道などの主要道路においては整備が進んでいるものの、住宅地や幹線道路ではまだまだ整備が追いついていない道路が多いのが現状である。事実、交通事故全体に占める歩行者を巻き込んだ事故の割合は、歩行者と車両の通行帯が完全に分離されている道路の多い欧米では10～15%であるのに比べ、日本では25%と比較的人身事故の割合が高くなっている。この様に狭く入り組んだ道路が多い日本では、交差点全てに路車強調型システムを設置することや、ましてや道路を新たに整備しなおすといった事は現実的に見て不可能である。

これらの点から路車協調型システムは交通量が多いなど特定の地点では設置が進んでいるものの、それ以外の地点では車載システムで解決するという見解が現在の主流になっている。

2.2 車載センサによる事故防止技術

車載センサによる安全運転支援技術についての研究は欧米でも古くから行われてきた。安全運転支援としては、障害物検知やレーン検出などの研究も存在するが、ここでは本研究の目的である歩行者検出技術に限って紹介する。

画像センサ以外を用いた歩行者検出技術の研究として、超音波センサやレーザーレーダ

ーを用いた手法[7][8]などが存在する。しかし画像センサは超音波センサやレーザーレーダーを用いる手法に比べ、画像自体から外見情報や動き情報また対象オブジェクトの軌跡など得られる情報が多いことから歩行者検出技術の研究では画像センサを用いての研究例が多い。しかし以下のように画像検出を用いた歩行者検出は難しい点がある。

- 当然の事ながら車両は移動するので、複雑な背景が不規則に動く中で歩行者を検出しなければならない。また天候、日の出日の入りなど時間によっても左右される事。
- 歩行者の出現パターンも千差万別である。物陰から突然でてくる場合もあれば歩行者同士が重なりあう場合もある。また、歩いていたり走っていたりとポーズや着ている服のテクスチャも人それぞれである事。
- 画像処理は通常大きなマシンパワーを要する、また安全運転支援を考えるとリアルタイム処理が求められるが、車載ゆえ用いるハードウェアに限界がある事。

などである。これら困難な条件の中世界各地で様々な研究が行われている。本章ではそれらの関連研究を紹介する。

2.2.1 外見情報を用いた歩行者検出技術の研究例

現在画像センサによる歩行者検出技術の研究例は、用いるベースとなる技術により大きく2つに分けられる。その1つが外見情報を用いた研究例である。

外見情報を用いるとは画像から特定の特徴量を抽出し、その特徴量を用いるという事である。特徴量自体の研究も多くなされているが、主に用いられている特徴量では Haar-Wavelet[9]や edge template[10], Histograms of Oriented Gradients(HOG)[11]特徴量などが挙げられる。Papageorgiou ら[9]や Oren[12]らは静止カメラで歩行者を前方や横後方から撮影し、撮影された画像から Haar-Wavelet 特徴量を抽出し、Support Vector Machine(SVM)判別器に入力する事で歩行者検出を行っている。Gavrilla ら[10]は Region of Interest(ROI)と呼ばれる歩行者候補領域を edge template を用いて抽出した後、ニューラルネットワークを用いて歩行者検出を行っている。また、Dalal ら[11]は HOG 特徴量と SVM を組み合わせている。Munder ら[13]は形状やテクスチャ、距離などを総合して特徴量とし、ベイジアンネットワークを用いて歩行者を検出したのち、パーティクルフィルタを用いて追跡を行っている。

2.2.2 動き情報を用いた歩行者検出技術の研究例

外見情報を用いた手法とは異なり、動き情報を用いた手法ではオプティカルフローによる短期的な歩行者の動きを検出して歩行者検出を行う。

Cutler ら[14]は歩行者の歩行の際の周期的な動きパターンに着目した。Sidenbladh ら[15]は歩行者の動きパターンと非歩行者オブジェクトの動きパターンを収集し SVM に学習させ、判別器を構成した。Curio ら[16]は検出された領域中足元と思われる領域の動きを判別器に掛けることにより、街中での歩行者検出を試みている。Viola ら[17]は外見情報も組み合わせる事により歩行者検出のモデルを構築した。彼らは *adaboost* を用いて動き情報、外見情報から有用な特徴量を探索し、その結果の特徴量を用いている。Elzein ら[18]はフレーム間の差異を利用して ROI を検出し、ROI 内のみオプティカルフローを計算している。さらに検出された ROI の内、衝突可能性が高い ROI のみに限り Haar-Wavelet 特徴量を用いて歩行者かどうかを判定した。

2.2.3 近年の歩行者検出技術の研究例

2.2.1 項や 2.2.2 に挙げた研究例は一見良い性能に思えるが、車両上で安全運転支援装置として適用する分にはまだまだ不十分である。近年ではこれらを組み合わせた例や、センサ自体を複数用いる研究例も存在する。

レーザーレーダーは 1.1 章でも述べた通り、空間分解能が低くその点で画像センサに劣る。しかしレーダー前方の障害物までの距離の測定に関しては、画像センサより高性能で速度も速い。このレーザーレーダーの利点に注目し Broggi ら[19]はレーザーレーダーにより自車両前方の障害物を検知し、画像センサを用いてその障害物が歩行者かどうかを判定する研究を行っている。また Seki ら[]は画像センサを二台用いてステレオシステムとして運用し、障害物までの距離マップを作成し検出対象を絞った後に、さらに Co-HOG 特徴量と SVM 判別器を組み合わせている。これらと同様の研究は他にも多くなされている[20][21]。

2.3 本研究の位置付け

[19]など、現状すでに車載システムとして実運用のレベルに達している研究例は存在している。しかしこれらの研究は全て自車両の前方に歩行者が飛び出すのを検知する事を想定している。つまり交差点などで自車両が曲がりながら、運転者の不注意により歩行者を巻き込む事故をカバーしていない。前方監視は周辺監視に比べ、車両の速度が一定で背景の動きも乱雑でなく、また歩行者以外の構造物も出現しにくいため技術的に容易である事が前方監視研究が多くなされている原因でもある。言い換えれば、交差点などでの巻き込み事故防止システムに実装できるほどのアルゴリズムは存在していないといえる。またセンサに関しても、高性能な研究はステレオシステムかフュージョンセンサを採用しているもののみで、実際に各個人の所有する車両に装着する事を考慮した場合、コストが非常に高くなってしまうことも問題である。

そこでこれらの問題を解決するために、本研究は第一に直線道路での前方監視も交差点での周辺監視も同一のプロセスで達成できるアルゴリズムを、さらに実システムへの運用を考慮に入れ、コストの低い単眼画像センサでの歩行者検出を目的にしている。

第3章

時空間 MRF モデルによるオブジェクト検出・トラッキング

時空間 MRF モデルは本研究室で開発されたオブジェクトトラッキング技術である。過去には事象検知技術と組み合わせた渋滞警報システムやカメラネットワークと組み合わせた広域監視システムにも応用された例がある。本研究では時空間 MRF モデルを、オブジェクト検出・トラッキング技術として用い、後半のオブジェクト判定と組み合わせて歩行者検出アルゴリズムとしている。

本章ではオブジェクト検出・追跡について説明する。

画像中の全てのピクセル上で動きベクトルが算出できる。本研究では、まずその動きベクトルの差異に注目し、背景と前景のオブジェクトを分離するモーションベースの物体検出手法を適用した。ここでのオブジェクトとは歩行者に限らない。動きベクトルが周囲とは異なる領域を前景のオブジェクトとして抽出し、これを歩行者の候補領域、即ち ROI(region of interest)とする。次に、ROI 内のテクスチャパターンを解析し、歩行者ではないオブジェクトと判定されたものを除外し、残った ROI を歩行者と見なす。本研究では以上の手法を提案する。以下ではまず動きベクトルを元にしたオブジェクト検出手法について述べる。なお、動きベクトルによるオブジェクト検出手法については前年度の研究成果を引き継いでいるため[21][22][23]、本章は引用が多く含まれている。

画像中の歩行者は、ビルの壁面などの構造物等で構成される背景面と動きベクトルが異なるため、動きベクトルの差異をもとに歩行者を検出することが可能である。また、交差点付近等で静止している歩行者であっても、カメラに対して遠距離にある背景構造物と近距離にある歩行者とでは、画像中での動きベクトル量が異なるため、モーションベースの歩行者検出が可能である。一方、この方法では電柱や建て看板、ポールや道路脇の樹木といった、カメラに対して背景の構造物より近距離にある前景オブジェクトも、歩行者と同様に検出されることになる。したがって、これらのオブジェクトと真の歩行者とを何からかの方法で識別することが必要であるが、これについては第4章で後述する。

一般に、カメラが動くことによる背景領域の動きは、なんらかの方法で定式化することが出来る。本研究では、直線道路および交差点での歩行者検出を目的にしているので、車載カメラは基本的に水平方向へ移動するものと仮定でき、背景物体のカメラとの遠近から、画像の水平方向に見た背景動きベクトル量の分布は、一次式（直線）で近似できると仮定する。

本研究では、画像の水平方向に x 軸を、鉛直方向に y 軸をとる。まず、 8×8 pixels の画

素集合からなるブロックを定義し、入力画像を分割する．図 3.1(a)に示すように、 y 座標が同一のブロックの集合をマッチングラインと定義し、このマッチングラインごとに、背景動きベクトルの一次式近似（直線回帰）を行い、これと設定された閾値以上に動きが異なるブロックを抽出する．

まず、設定されたマッチングラインについて、各ブロックの動きベクトル $M(x, y)$ を求める．ここで、 (x, y) は対象ブロックの座標を表す．動きベクトル $M(x, y)$ は、通常のブロックマッチング法によりピクセル単位で求められる．本研究における探索範囲は、 x 軸方向に ± 24 ピクセル、 y 軸方向に ± 5 ピクセルと設定した．背景画像が水平方向に移動することを仮定しているため、水平（ x 軸）方向の動きベクトルを基本としている．したがって、 $M(x, y)$ は、ブロックマッチング法で求められた動きベクトルのうち、 x 軸方向の成分を表すスカラー量である．ただし、鉛直（ y 軸）方向はカメラの上下振動の影響を吸収することを目的としたに探索を行なうため、 y 軸方向の探索範囲は x 軸方向に比べて狭いものとなっている．

図 3.1 (b)は、図 3.1 (a)のマッチングラインに対して、動きベクトル探索を行なった結果である．ただし、あるラインが設定されれば、対象ブロックの y 座標はマッチングラインに適応した定数値が入るため、 $M(x)$ と表記を省略している．図 3.1 のスキャンライン上の画像はビル背景によるものが大勢を占めているため、動きベクトルは、遠景から近景にかけて、ほぼ直線状に分布することが考えられるが、必ずしも動きベクトルを算出できるとは限らない．これは、領域内のテクスチャが複雑であったり、繰り返しパターンで構成されていたりという場合に、ブロックマッチングにより動きがマッチする箇所が複数存在する可能性があるからである．そこで、これらのエラーを修正して、正しい動きベクトル分布を得るため、次のようなアルゴリズムを適用した．

【動き量の最適化アルゴリズム】

Step.1: 設定されたマッチングラインに属する全てのブロックについて、ブロックマッチングにより動き量 $M(x)$ を求める．ここで、ラインを設定した時点で y 座標は固定されているため、 $M(x, y)$ から y を省略している．

Step.2: 最小二乗法により、 $M(x)$ の回帰直線 $L(x)$ を求める．

Step.3: 全てに x について、 $M(x)$ が $M_{line}(x)$ と比べて閾値以上異なる場合は、 $M(x)$ が

$M_{line}(x)$ およびその近傍値をとるものと仮定する．当該ブロックに対して、当該仮定した動きベクトルに対応した動き量に基づく画素値の絶対差分を求める．絶対差分値が閾値以下であれば、当該仮定した動きベクトルを $M(x)$ として採用する．

Step.4: Step.3 において、 $M(x)$ に修正が行われた場合は、Step.2 に戻る．修正が行われな

かった場合は Step.5 に進む.

Step.5: $M(x)$ を $M_{opt}(x)$ として, $L(x)$ を $L_{opt}(x)$ として登録し, 最終決定する.

[アルゴリズム終了]

$M(x)$ と $M_{opt}(x)$ の関係を図 3.1(b)(c)に示す. 図 3.1(b)においては, 多くのエラーが発生しているが, 図 3.1 (c)では, エラーがほぼ修正されていることが分かる.

図 3.2 は, 図 3.1 と同じ画像について, 歩行者が存在する高さ (y 座標) において, 動き量のラインマッチングを行った結果である. 図には, $L_{opt}(x)$ と $M_{opt}(x)$ との差異が背景の動きと異なる前景のオブジェクトとして表れていて, それぞれ歩行者や壁のくぼみを表している. これらの差異がある部分をオブジェクトとして検出し, その領域情報が時空間 MRF モデルへ入力される.

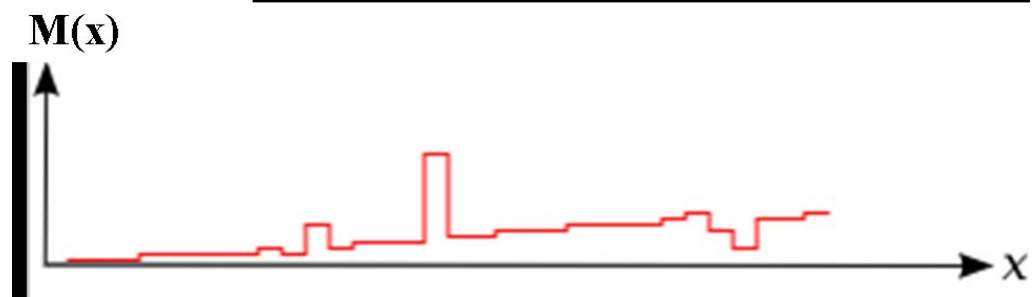
図 3.3 の各シーン上段は, 上述した動きベクトル検出方法により, 背景と動きベクトルが異なると判断されたブロックを示したものである. このように, 背景画像の動きと異なる部分は, ブロック単位で求められる. これらの検出されたブロックに対し, 隣接するもの同士をクラスタリングする. 各クラスタに属するブロックにクラスタ固有の番号を付与したものをオブジェクトマップとし, 時空間 MRF モデルに当該オブジェクトマップ情報を入力する. この同一の番号を付与された領域が一つの ROI に相当する.

時空間 MRF モデルは, 受け取ったオブジェクトマップの情報により, アルゴリズム内部のオブジェクトマップを初期化し, 以後, 時空間画像の領域分割を行う. 図 3.3 各シーン上段で得られたクラスタ情報をもとに, 時空間 MRF モデルがトラッキングを行った結果を図 3.3 下段に示す. ここで, トラッキングとは時空間画像の領域分割と等価であり, 図 3.3 下段は理解のため, トラッキング結果を領域マップの外接長方形で示している.

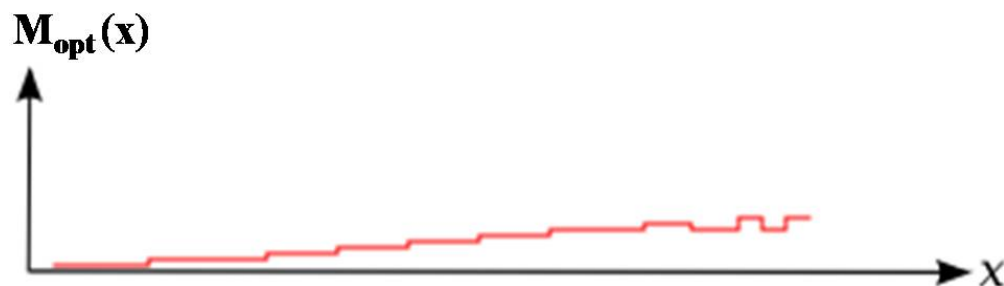
なお, 時空間 MRF モデルについての詳細な説明は[]を参考にされたい.



(a) 入力画像とマッチングラインの例



(b) 初期エラー状態

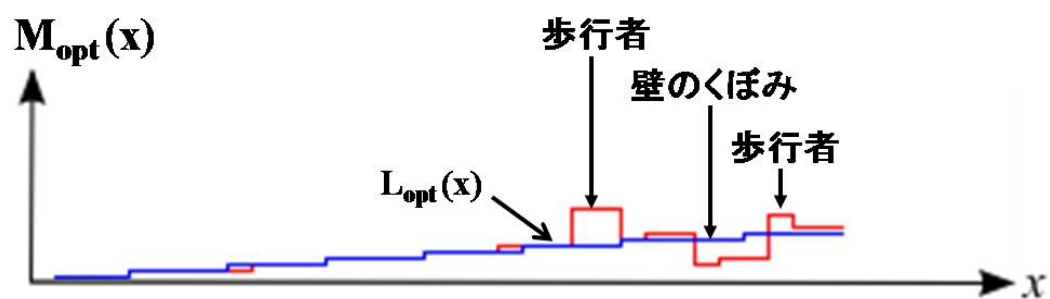


(c) 回帰直線を参照し最適化された状態

図 3. 1 動きベクトル初期エラー状態と最適化状態



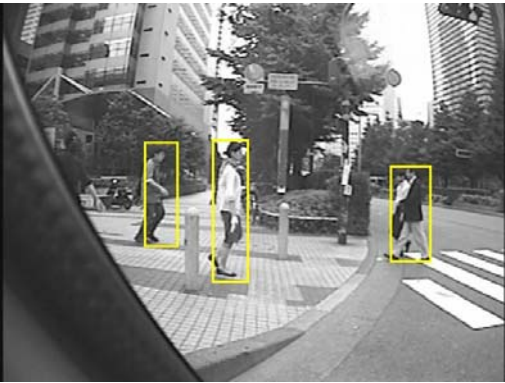
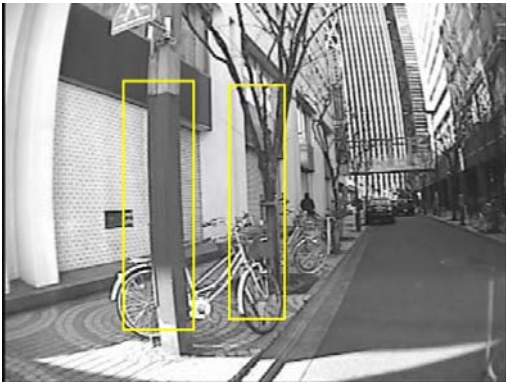


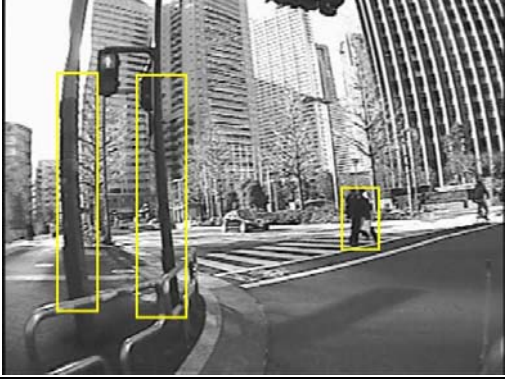



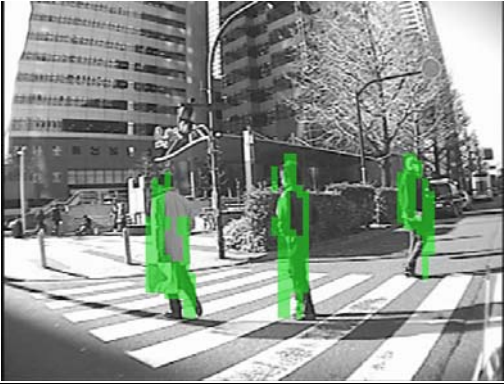
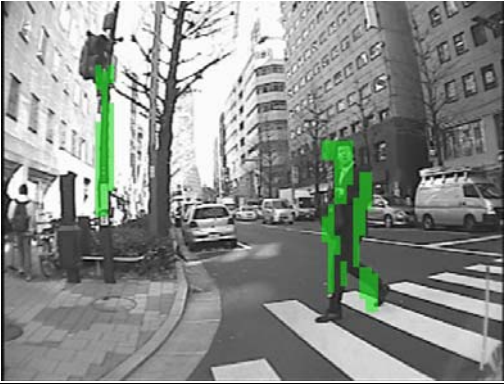
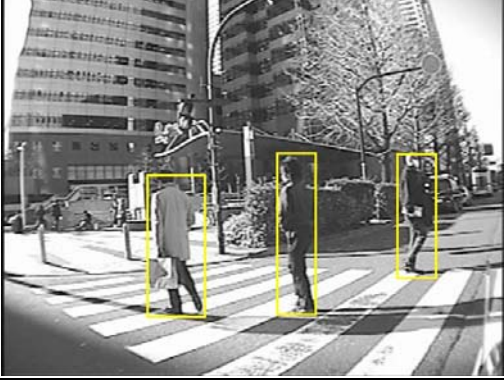
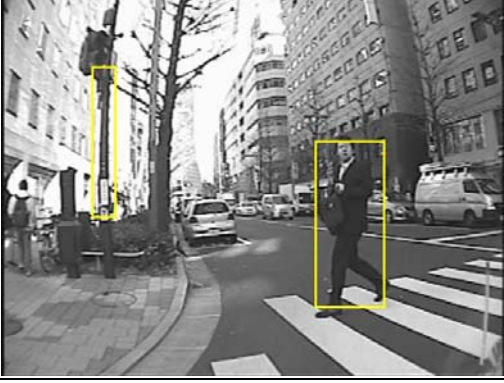
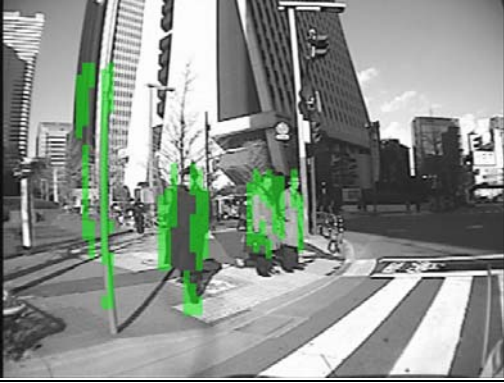
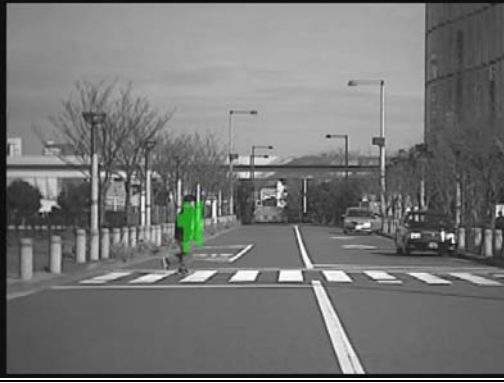
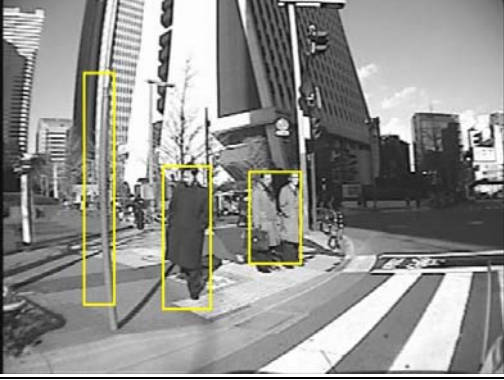
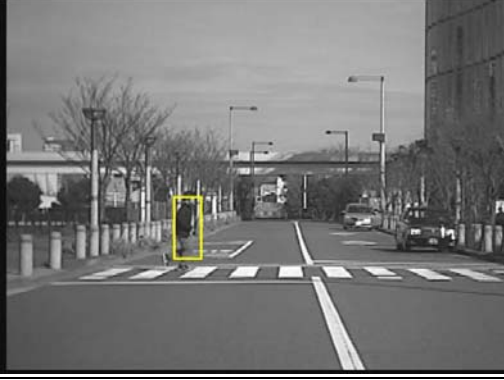
(a) 入力画像とマッチングラインの例



(b) 動きベクトルの回帰直線と動き差分

図 3. 2 ブロックライン上での動きベクトル最適化

	
	
<p>シーン 1</p>	<p>シーン 2</p>
	
	
<p>シーン 3</p>	<p>シーン 4</p>

	
	
<p>シーン 5</p>	<p>シーン 6</p>
	
	
<p>シーン 7</p>	<p>シーン 8</p>

	
	
<p>シーン 9</p>	<p>シーン 10</p>
	
	
<p>シーン 11</p>	<p>シーン 12</p>

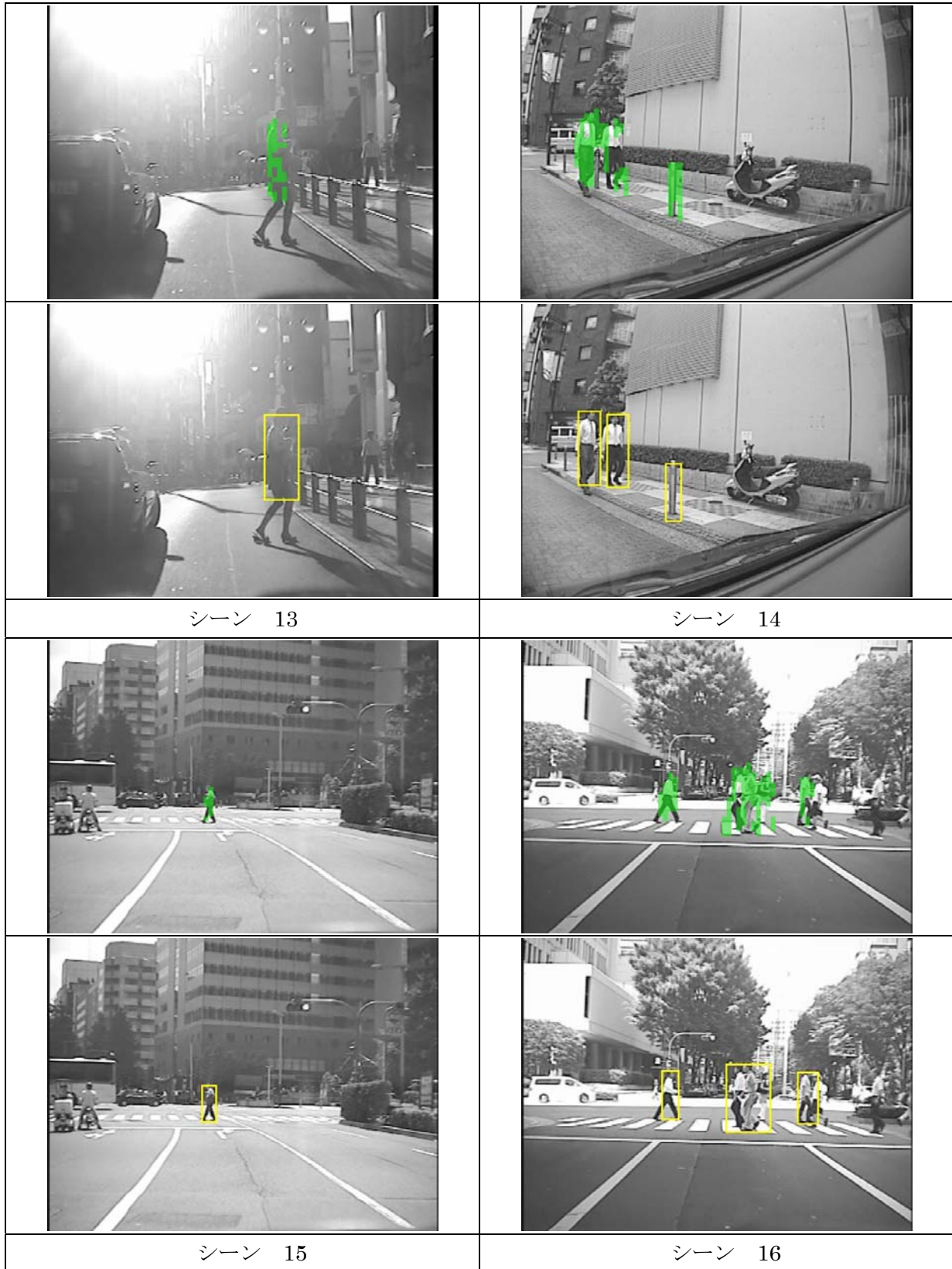


図 3.3 時空間 MRF モデルによるオブジェクト検出結果

第4章 オブジェクト判別

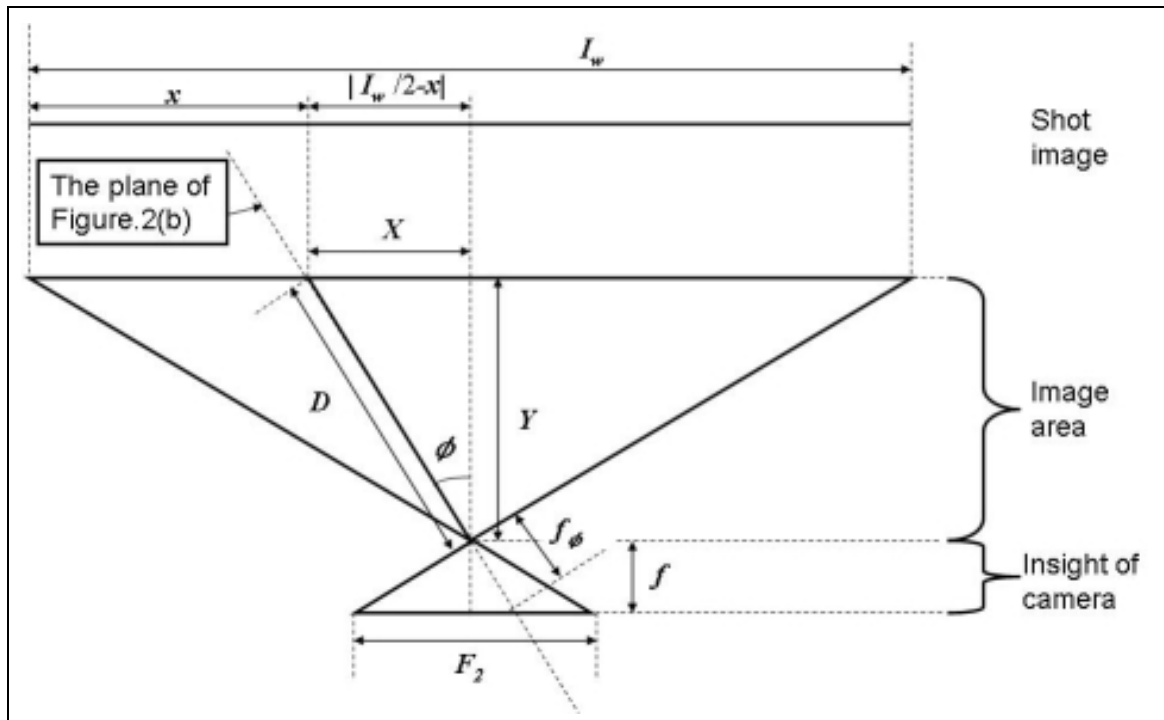
前章では時空間 MRF モデルによるオブジェクト検出・追跡の技術について説明した。オブジェクト判別段階では検出された ROI から特徴量を抽出し、判別器に入力し歩行者かどうかの解を得るが、本章ではそのプロセスおよび ROI の幾何拘束について説明する。

4.1 幾何拘束によるオブジェクト選別

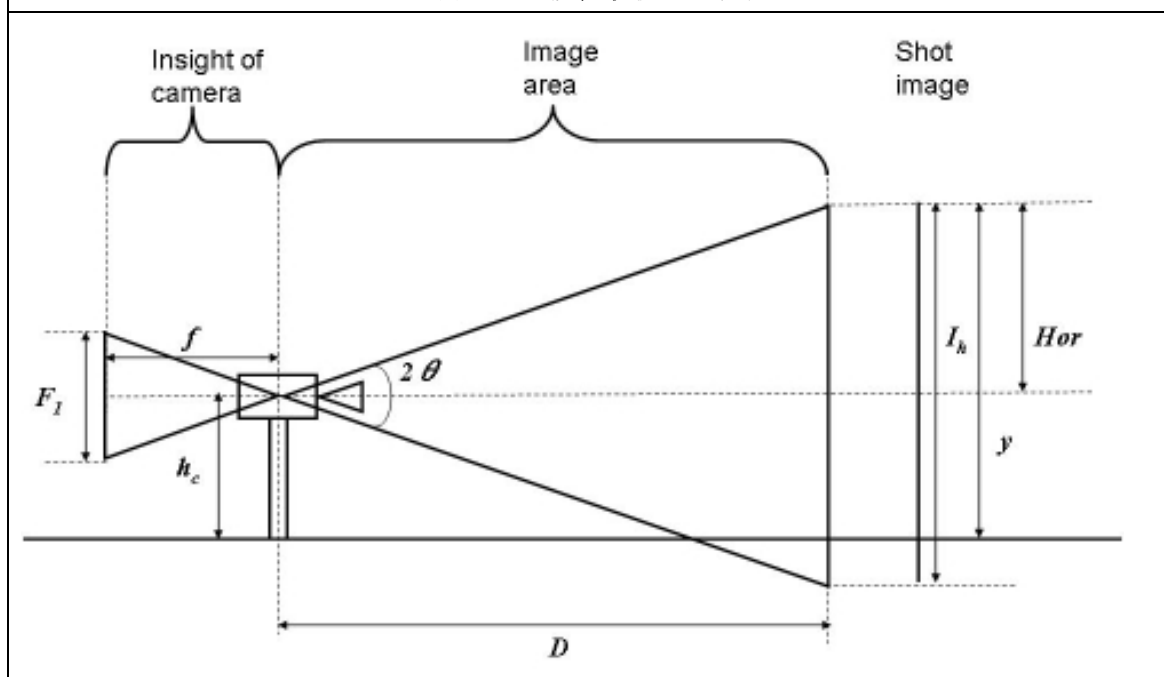
第3章の時空間 MRF モデルにより検出されたオブジェクトは、本章で述べるオブジェクト判定アルゴリズムにより歩行者とそれ以外に分別される。しかし時空間 MRF モデルにより検出されたオブジェクト総数が多数である場合、オブジェクト全てに対しパターン識別の処理を行う事は計算コストが増大すると共に、誤報の増加を招いてしまう恐れもある。よってそれらの問題を回避する為に、パターン識別処理に入る前段階で検出されたオブジェクト ROI の位置と高さから歩行者である確率が低いと思われるオブジェクトを識別対象から除外する操作を行った。

4.1.1 オブジェクト ROI の高さ推定

本項では、画像上でのオブジェクト ROI 情報から現実でのオブジェクト ROI の高さを推定する際に必要なカメラキャリブレーションについて述べている。



(a)カメラ設置状況の上面図



(b)(a)中斷面上の側面図

図 4.1 シーン撮影時のカメラ設置状況

図 4.1 は本研究におけるシーン撮影時カメラの設置状況を示している。また、本研究では、最も人身事故が起こり易い交差点と直線道路上での人身事故での歩行者追跡を目的にしているため、したがってカメラの光軸が水平に設置されていると仮定できる。図 4.1(a)は設置状況の上面図を、図 4.1(b)は(a)中で示されている断面の側面図を示している。

“ f_ϕ ”は図 4.1 (a)中の “ ϕ ”に伴い変化するように見えるが、レンズ特性によりレンズの焦点距離 “ f ”の一定値となる。これにより現実でのカメラからオブジェクトの距離と、画像上でのオブジェクトの高さの関係が導き出せる (式(4.1))。式(4.1)でカメラからオブジェクトの距離は“ D [メートル]”で、現実での 1 メートルが画像上で何ピクセルに相当するかを“ h_p [ピクセル / メートル]”で表している。カメラ画角 20 度, 30 度, 60 度, 75 度, 100 度での“ D ”と “ h_p ”の関係が図 4.2 に示されている。

図 4.2 で“ $[x,y]$ ”は画像上での ROI の底辺の座標を表しており、これより現実での道路上の座標である“ $[X,Y]$ ”に変換できる(式(4.2))。よって距離“ D ”が画像上の座標“ $[x,y]$ ”によって表すことができる(式(4.2))。その関係を示したものが図 4.3 である。

最終的に、式(4.1)と式(4.3)から画像上座標“ $[x,y]$ ”と 1 メートル当たりのピクセル数“ h_p ”の関係を表す式(4.4)が、更に式(4.4)から、現実でのオブジェクトの高さ“ H ”を表す式(4.5)が導き出せる。従って上記から ROI の画像上での座標“ $[x,y]$ ”と高さ“ h ”が得られれば、現実での高さ“ H ”も推定できる。

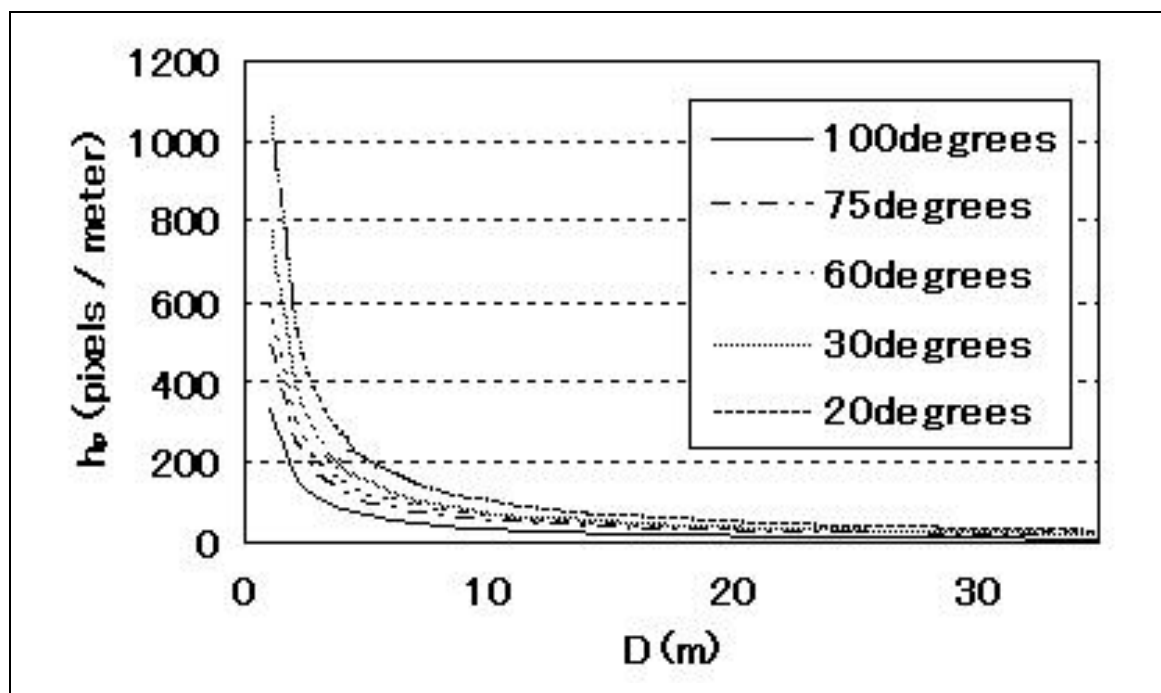
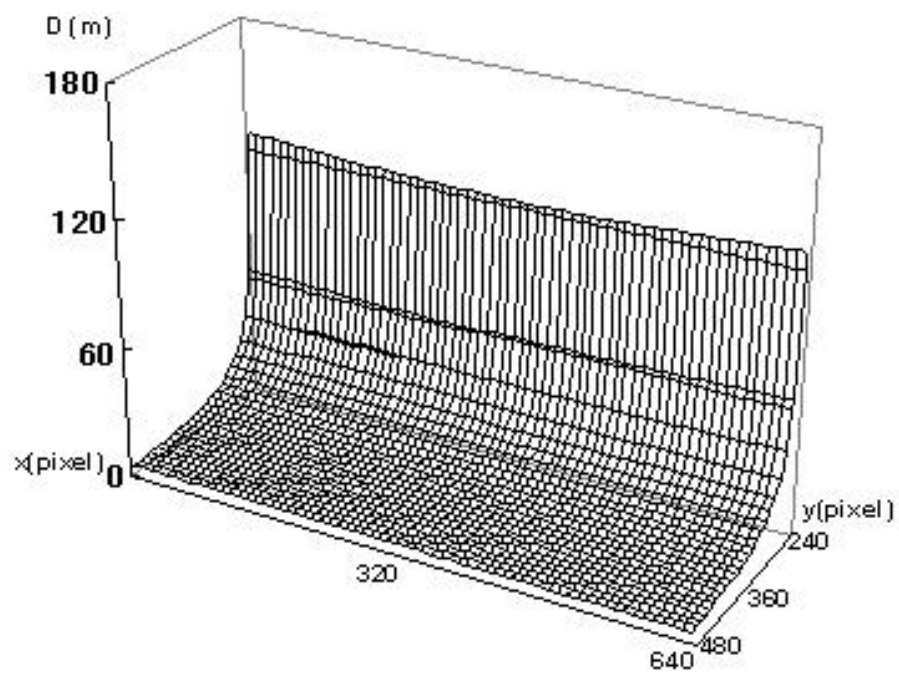
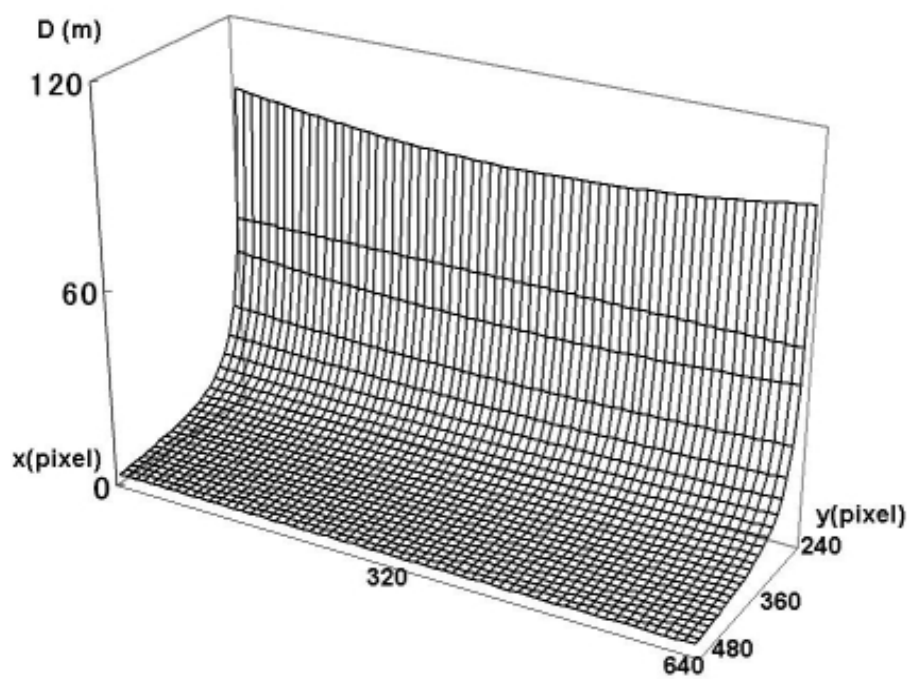


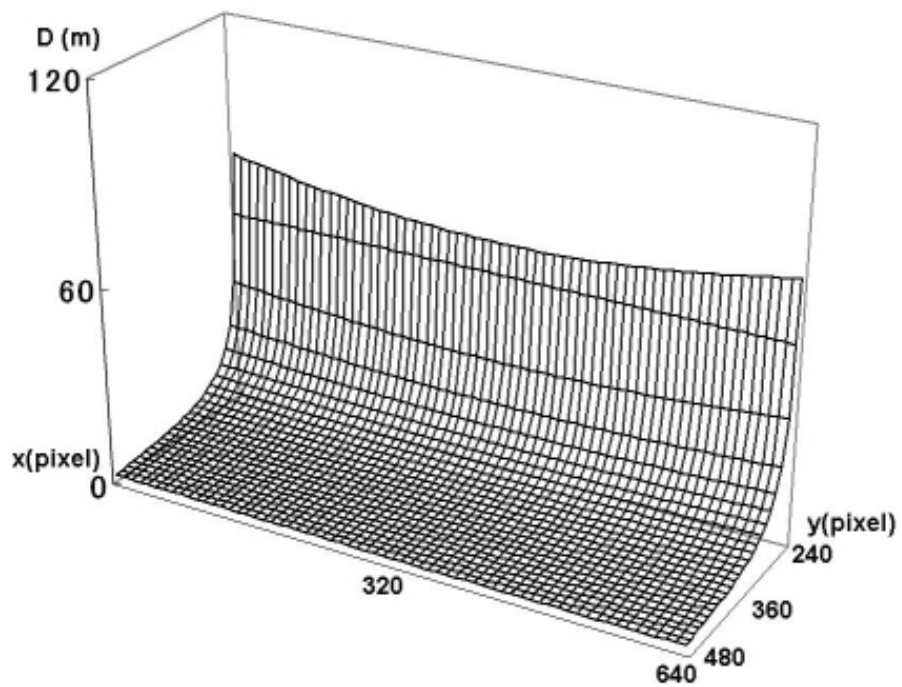
図 4.2 1m 当たりのピクセル数 h_p [pixels / meter] と カメラからの距離 D [meters]



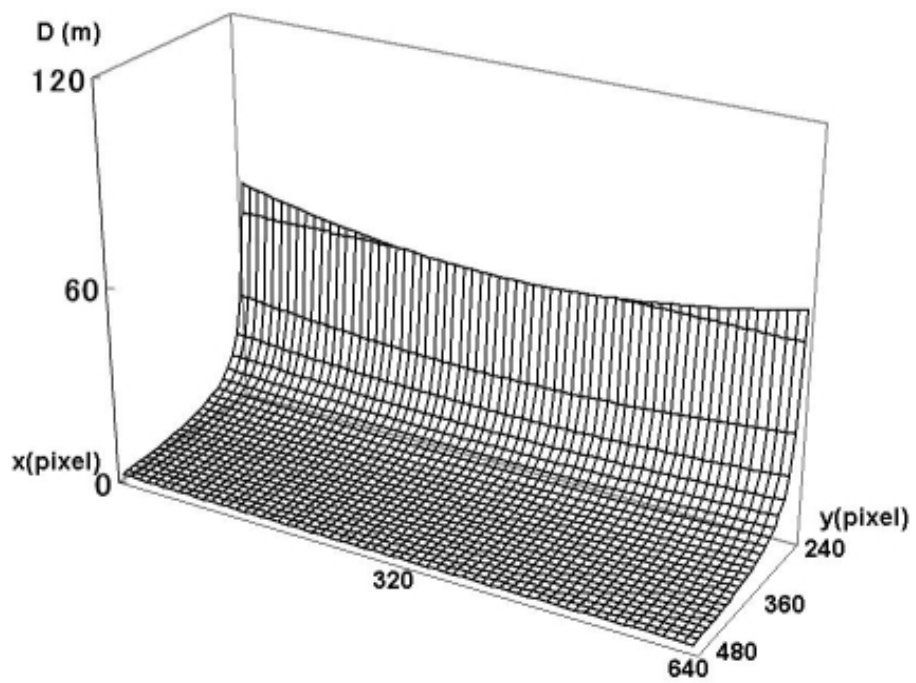
20 度



30 度



60 度



75 度

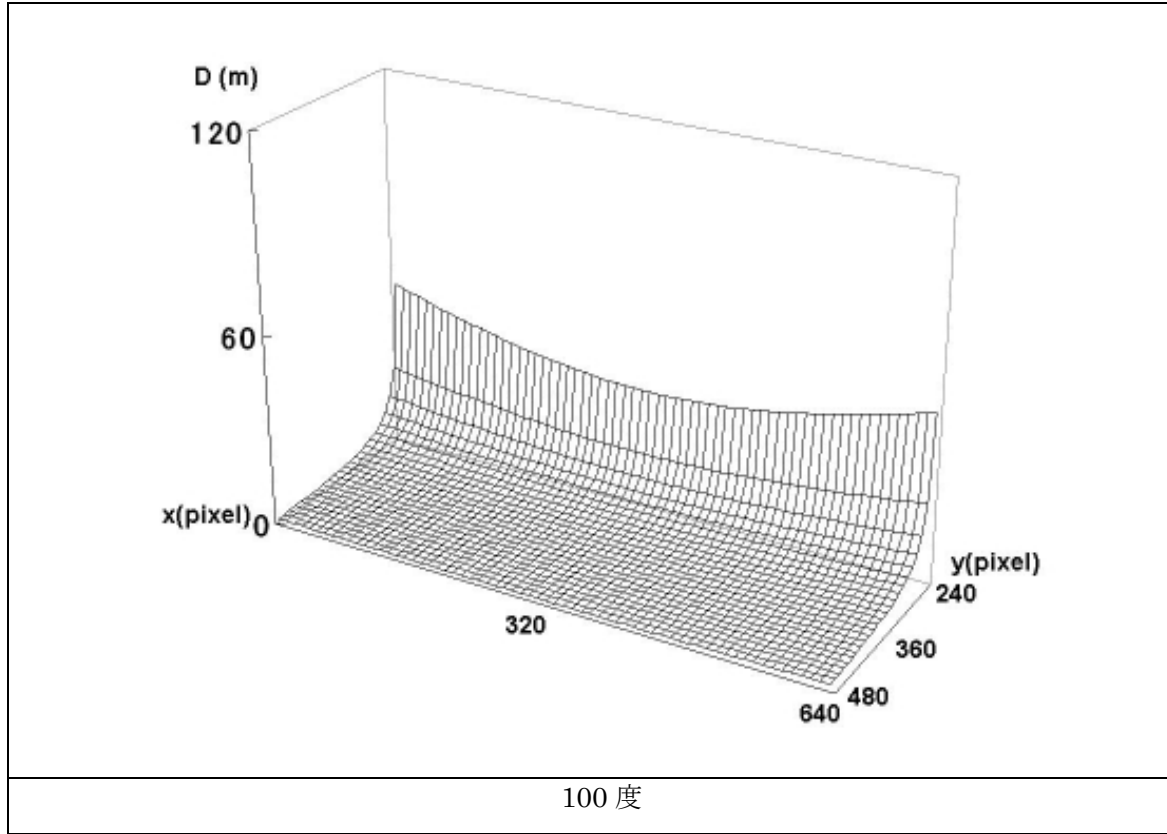


図 4.3 位置(x, y) [pixels]とカメラからの距離 D [meters]

$$h_p = \frac{I_h f}{F_1 D} \quad (4.1)$$

$$\begin{cases} X = \frac{F_2 Y}{I_w f} |I_w / 2 - x| \\ Y = \frac{2 f h_c}{F_1} \times \frac{Hor}{Hor - I_h + y} \end{cases} \quad (4.2)$$

$$D = \sqrt{X^2 + Y^2} = \frac{2 f h_c}{F_1} \times \frac{Hor}{Hor - I_h + y} \times \sqrt{1 + \left(\frac{|I_w / 2 - x|}{I_w} \times \frac{F_2}{f} \right)^2} \quad (4.3)$$

$$h_p = \frac{I_h(Hor - I_h + y)}{2h_c Hor \sqrt{1 + \left(\frac{|I_w/2 - x|}{I_w} \times \frac{F_2}{f} \right)^2}} \quad (4.4)$$

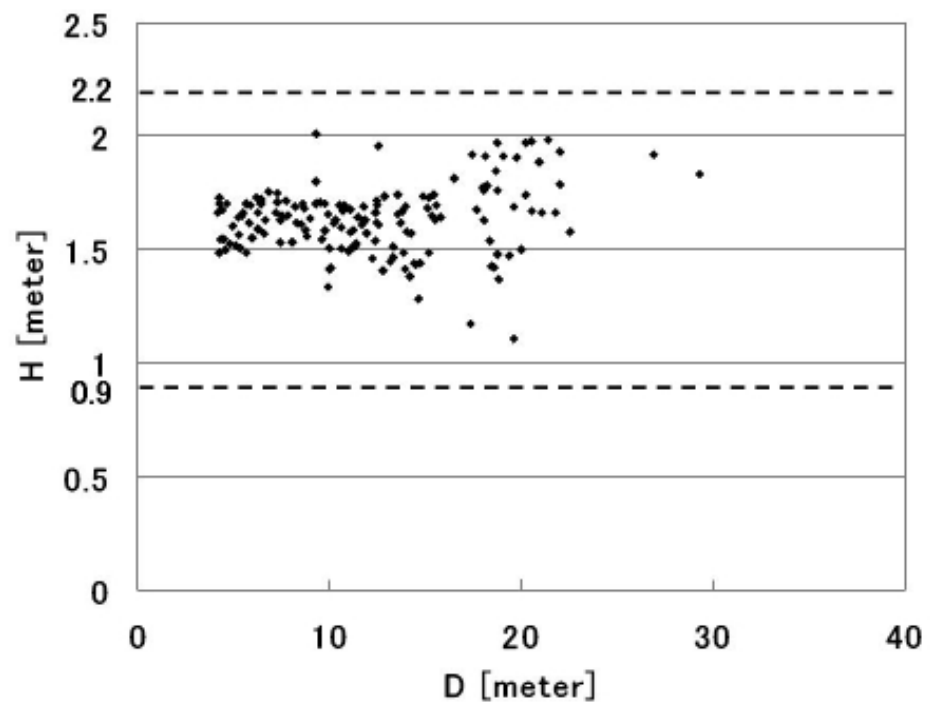
$$H = \frac{h}{h_p} \quad (4.5)$$

4.1.2 ROI 高さ “ H ” の検証

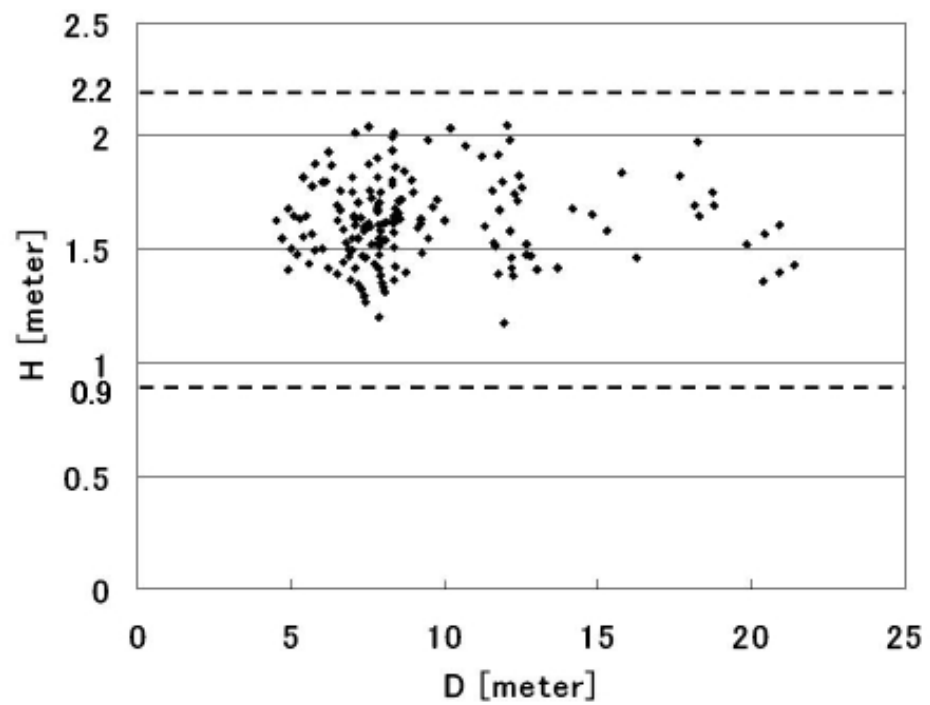
前項の計算により ROI の画像上座標“ $[x,y]$ ” と高さ“ h ”から実世界上の高さ“ H ”が導き出せた．これにより各 ROI に“ H ”が与えられる．そしてこの“ H ”が設定された閾値以上か以下の ROI は歩行者として高すぎるか低すぎるとされ，後のパターン識別での対象から外れることになる．ここではその閾値を決定するために“ H ”の偏差について実験を行った．

交差点での右左折時，車体は遠心力によって左右に沈み込む．また直線道路走行時にも加減速による沈み込みがあると考えられる．その沈み込みによってカメラの光軸の水平位置からのズレが発生する．このズレは ROI の画像上での座標“ $[x,y]$ ”に影響し，結果として距離“ D ”や“ h_p ”，“ h ”にまでも光軸が水平だった場合からのズレを引き起こすことになる．さらに“ h_p ”，“ h ”がずれることによって推定された歩行者の高さ“ H ”もずれてしまうことになる．しかし実システムでの運用を考慮した場合，車体の沈み込みによる光軸のズレを計測し，“ $[x,y]$ ， D ， h_p ”を修正するのは困難である．よって本項では実際の道路を走行してデータを収集し，データから直接これらのズレを許容するような“ H ”の閾値の上限と下限を設定した．

本研究では，カメラの内部パラメータによる各個体間でのズレは沈み込みによるズレに比べ非常に小さいと考え，これらを計算により求めることは行っていない．加えてこの段階で厳格に ROI を選別する事は，後段のパターン識別に掛けられる対象を減少させ，検出率の低下を招く恐れがあるため，“ H ”の閾値を厳密に設定する事は行っていない．後述する図 4.4 は実際の街中を走行し収集された撮影データを用いて，カメラから歩行者までの距離“ D ”と歩行者の身長“ H ”を算出し，グラフ上にプロットしたものである．このプロットした点を許容する範囲として 0.9m ～ 2.2m の閾値を設定した．これらの図から閾値内に点が収まっているのがわかる．



30 度



60 度

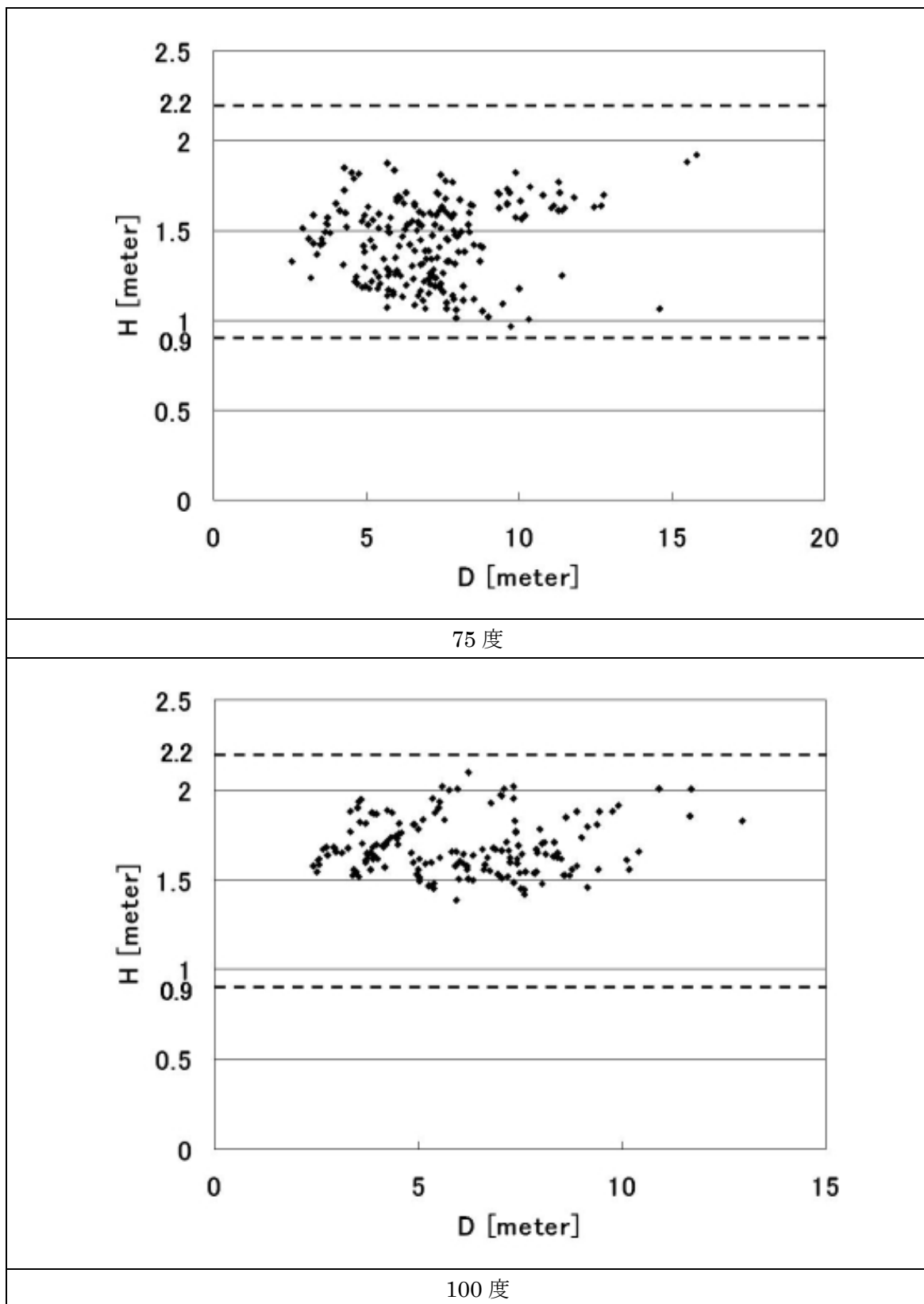


図 4.4 収集されたカメラからの距離 D [meter]に存在する人の身長 H [meter]

4.2 HOG カスケード判別器

本研究におけるシステムデザインとしては、車載カメラによるモーションベースの物体検出およびトラッキング手法により歩行者やその他構造物を検出した後に、判別器を用いて歩行者検出を行うといったアルゴリズムを構築している。これによりパターン識別のみに頼るところが大きい従来研究より、安定かつ誤報の少ないシステムが構築されると期待される。本項では Histograms of Oriented Gradients(HOG)特徴量を用いたカスケード型判別器の詳細について述べる。

4.2.1 HOG 特徴量について

現在まで数多くなされてきた歩行者検出技術の研究であるが、それらの内多くの研究が用いる特徴量として HOG 特徴量を用いている。また、学習判別器として SVM (サポートベクターマシン : Support Vector Machine) や線形 Fisher 判別器などを組み合わせることによって、オブジェクト判別分野に高い成果を挙げている。本項では本研究に用いられている HOG 特徴量の詳細について、本研究における HOG 特徴量の抽出手法について述べる。

HOG 特徴量一言で表すと、対象領域のエッジの方向の分布を表した特徴量だと言える。現在ではオリジナルの HOG 特徴量を改良または拡張した特徴量も数多く存在しているが、HOG 特徴量を最初に提言した原著では以下のように算出している。

まず対象領域中の 8×8 ピクセルの矩形領域を 1 ブロックとして定義する。次にブロック内の 8×8 の 64 画素についてその画素のエッジ方向を 9 方向に量子化して算出する。なお、ここで言うエッジとは輝度勾配を表す。その算出された値を 64 画素についてヒストグラム化する。するとこのヒストグラムはそのブロックが 9 次元のどの方向にエッジが強いかを判定できるベクトルに変換できる。しかし、エッジの強さは光量やエッジを構成する前景と背景のコントラストなどによって変化変動してしまう。それらの影響を吸収するために、抽出したベクトルに対し正規化を施す。あるブロックに関して当該ブロックを含む近傍 4 ブロックで構成される 16×16 ピクセルの領域を正規化領域とし、前述のヒストグラムを 4 ブロック分つなぎ合わせ 36 次元の特徴量を得る。この正規化領域を 128×64 サイズの画像上で 8 ピクセル(1 ブロック)ずつラスタスキャンさせてゆくと 7×15 回正規化領域を算出することになる。 128×64 サイズの場合、最終的に 105 の正規化領域の特徴量をつなぎ合わせ 3780 次元の特徴量が 1 枚の画像から抽出される。以上のようにして画像から HOG 特徴量を抽出する。本研究ではオリジナル HOG 特徴量にアレンジを加えたものを用いているが、

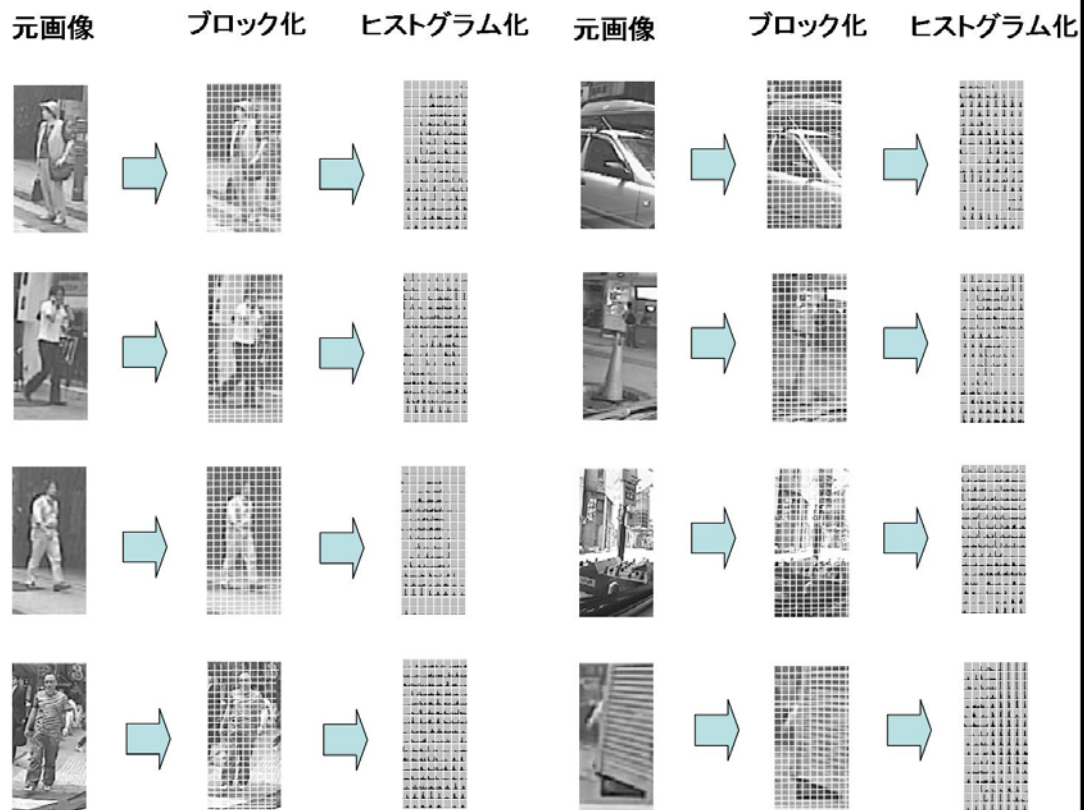


図 4.5 HOG 特徴量をヒストグラム化した画像

それについては後述する．図 4.5 はある画像と，ある画像から HOG 特徴量を抽出しそれをヒストグラム化した画像である．画像列左列が元画像，中心列がブロック化した画像，右列がベクトルのヒストグラムを画像として視覚化したものである．HOG 特徴量はエッジの強さをベクトル化した特徴量なので，右列を見るとエッジが存在しない領域（人の周りの一面灰色の領域など）には HOG 特徴量のベクトルがほとんど存在していない事が判る．

以上のように画像から特徴量を抽出し，それらを学習データやパターン識別の際に用いている．

4.2.2 学習データ構成

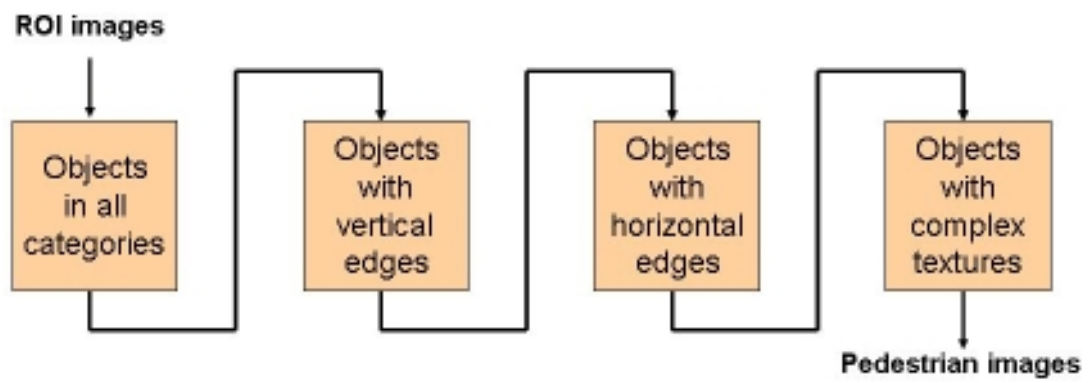
通常，機械判別とは 2 クラス判別に用いられる．本研究ではその 2 クラスを歩行者と非歩行者としている．機械判別について手短に説明すると，ある特徴量を用いて対象データを特徴量空間に写像し，その空間を分割する超平面を学習データを用いて生成し，判別したいデータが超平面のどちら側に属するかでそのデータのクラスを決定する，というものである．この超平面をどのように生成するかを学習データを参考にして決定する．超平面

の生成法は多様に存在するが、本研究では実システムを運用する場合を考慮し、計算コストの観点から線形判別器である **Fisher** 判別器を用いている。

さて、**HOG/Fisher** 判別器を用いてあるデータを歩行者か非歩行者かを判別したい訳であるが、通常非歩行者データは **HOG** 特徴量空間に広く分布しているので、歩行者データと非歩行者データを単一の線形超平面で区切ることは難しい。なぜなら歩行者に比べ非歩行者とはポール、看板、影、樹木、車など様々に存在するため、特徴量空間でもそれらがそれぞれ重なりあいながらクラスタを形成しているからである。そこで本研究ではそれぞれ独自の超平面をもつ判別器をカスケード接続し、歩行者非歩行者が入り混じったクラスタから少しずつ非歩行者を削ぎ落としていくようなイメージで判別に当たった。このような手法は従来研究でも広く用いられている。

カスケードの構成については図 4.6(a)に示す。本研究では、そのシンプルさと線形 **SVM** と同等の性能を持つことから、カスケードの各段に **Fisher** 判別器を用いている。カスケードは全部で4段である。各段では入力されたデータが非歩行者と判別された場合、データが除外され次段には入力されず、歩行者と判別されたデータのみが次段の判別器に入力される。そして最終段の判別器を通過したデータが歩行者とされる。

図 4.6(b)では判別器の各段で用いた学習データを示す。歩行者データとしては、従来の歩行者検出技術分野で広く用いられている **INRIA**[24]のデータベースから 1399 枚(**INRIA**)と、本件吸湿が東京の都心などを実際に走行して収集したシーンから抽出した 1299 枚(**KMJ**)を用いた。また、非歩行者データに関しても独自に収集した走行シーンから抽出したものを使用した。図 4.6(c)に非歩行者学習データを示す。まず初段では、2663 枚の実際の走行シーンに登場する様々な非歩行者データを使用した(**All Category**)。これはまず後段の処理コストの低減などを目的に大雑把に非歩行者データを除外するため初段をこれに設定した。そして初段を通過して誤判定されたデータを分析すると、建物や前方を走行中の車体などのエッジが強く出ているデータが非常に多いことが判明した。よってカスケード2段目と3段目では、それぞれ垂直方向に強いエッジを持つものと水平方向に強いエッジを持つもので構成した(**Horizontal Edge, Vertical Edge**)。そしてそれでも残ったデータを分析すると、樹木や街の雑踏、自転車の一部など非常にエッジが込み入っているデータが多かったため、最終段はそれらを強く学習させた(**Complex Textures**)。



(a) カスケード構成

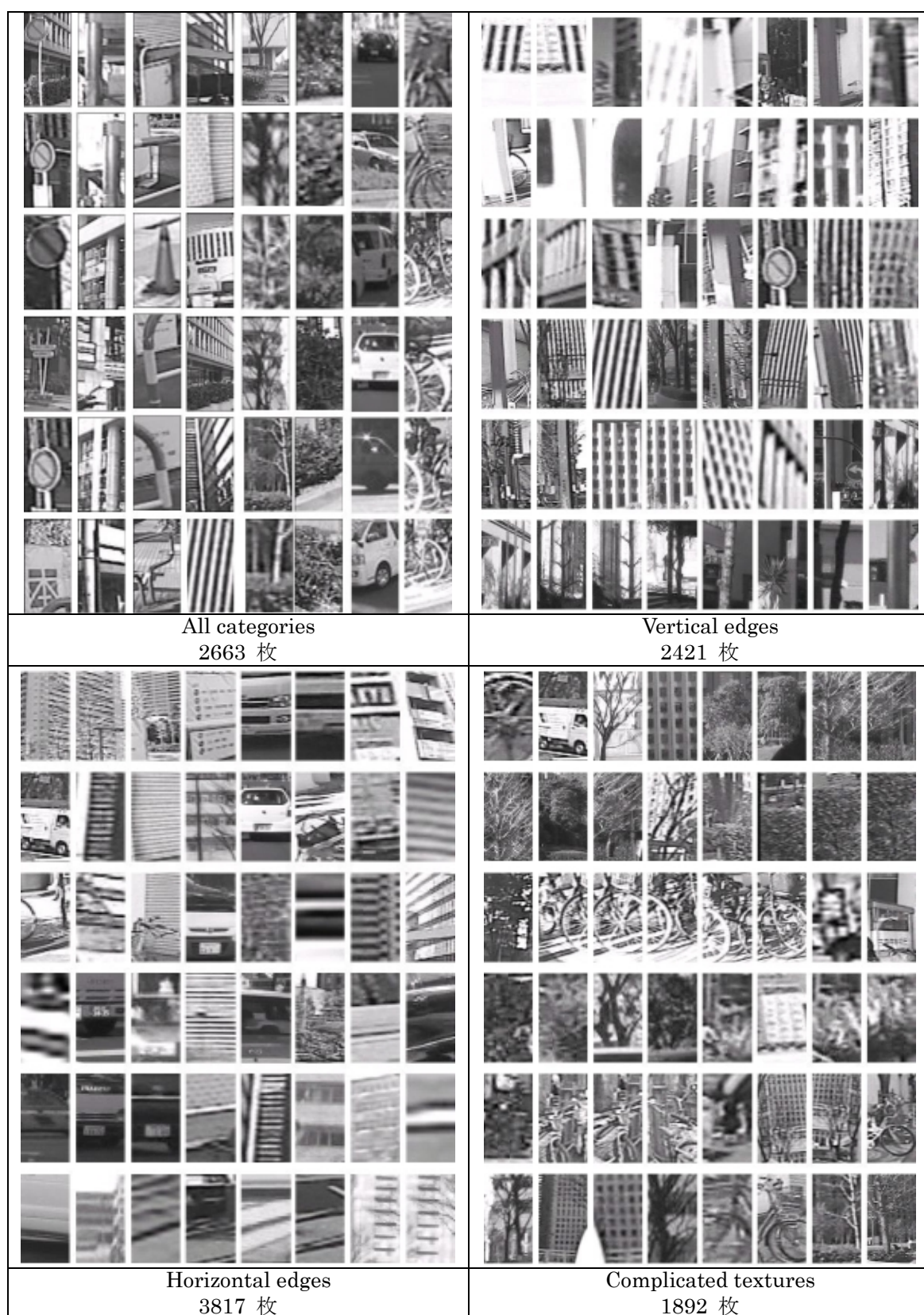


KMJ : 1212 枚



INRIA : 1399 枚

(b) 歩行者学習データ



(c)非歩行者学習データ

図4.6 学習データ構成

4.2.3 判別器の性能検証について

本研究で前項に、判別器を計算コストの面から線形判別器である **Fisher** 判別器を用いていると述べたが、その他の判別器についても検証を行った。

SVM については従来研究においても広く用いられており、また **Fisher** に比べて学習データの欠落に強いロバスト性を持つとして知られる一方で、**Fisher** は **SVM** に比べて計算コストが低いと事がメリットとして知られている。しかし、**Fisher** は用いる次元数の3倍数程度の学習データ数がなければ判別性能を十分に発揮できないといった点も挙げられる。

4.2.1 項で述べた原著で紹介されている **HOG** 特徴量は、正規化の際に1ブロック9次元の **HOG** 特徴量を近傍4ブロックについて結合し、計36次元の特徴量として扱っていた。アルゴリズム上は同一ブロックについて4回計算されていることになり、この重複部分が明度の変化などへのロバスト性を強くする。しかし本研究ではノイズに対するロバスト性は充分と考え、この重複部分を削除し1ブロック当たり1回のみ計算することで計算コストの削減を図った。

図4.6に示す学習データを用いて **SVM** と **Fisher** 判別器の歩行者非歩行者クラス判別性能の評価実験を行った。結果を表4.1に示す。**SVM** は36次元のオリジナル **HOG** を用いて判別した場合は **Fisher** に比べ高い性能を示すことがわかった。しかし、本研究が採用している9次元の **HOG** 特徴量を用いた場合、ほぼ同等の性能を示すことも判明した。よって本研究では9次元 **HOG** 特徴量を採用した。

表4.1中の学習データ判別性能とは、学習に用いたデータを判別する性能を示し、テストデータ判別性能とは学習に用いていないデータに対する判別性能を表す。また表中の計算時間や判別性能については309枚のテスト画像について5504回の判別を行った結果である。

表 4.1 SVM と Fisher の判別性能比較結果

判別器	HOG 特徴量 次元数	総次元数	学習データ判別性能(%)	
			歩行者	非歩行者
SVM	36	3780	97.16	97.93
	9	1152	92.54	94.33
Fisher	36	3780	53.12	56.54
	9	1152	98.92	97.91

判別器	テストデータ判別性能(%)		計算時間(s)
	歩行者	非歩行者	
SVM	87.85	91.92	
	90.00	84.72	8.2
Fisher	77.00	76.20	
	89.00	86.68	0.41

4.3 セマンティック HOG 特徴量による更なる選別

本研究は前項までの説明にあるとおり，オブジェクト判別の段階ではカスケード型 HOG/Fisher 判別器を用いている．この手法は特徴量の雲を線形な超平面で徐々に区切っていくようなものであるが，これではどうしても歩行者のクラスに混じってしまう非歩行者オブジェクトが存在してしまう．これは線形の超平面では歩行者と非歩行者のデータが入り混じっている領域を完全に分離することができないからである．そこで非線形な判別器を用いて複雑な超平面で分離する手法が考えられるが，非線形な判別器では効果にあまり差がないのに比べ処理コストがかかってしまうのは前述の通りである．

線形判別器がどうしても歩行者と判断してしまう非歩行者オブジェクトを分析した結果，非常に強い特徴を持っている非歩行者オブジェクトが存在していることがわかった．そこで HOG 特徴量を用いてそれらの特徴を強くするような新たな特徴量を作成し，それをもちいて，代表的な特徴をもつオブジェクトの排除を試みた．

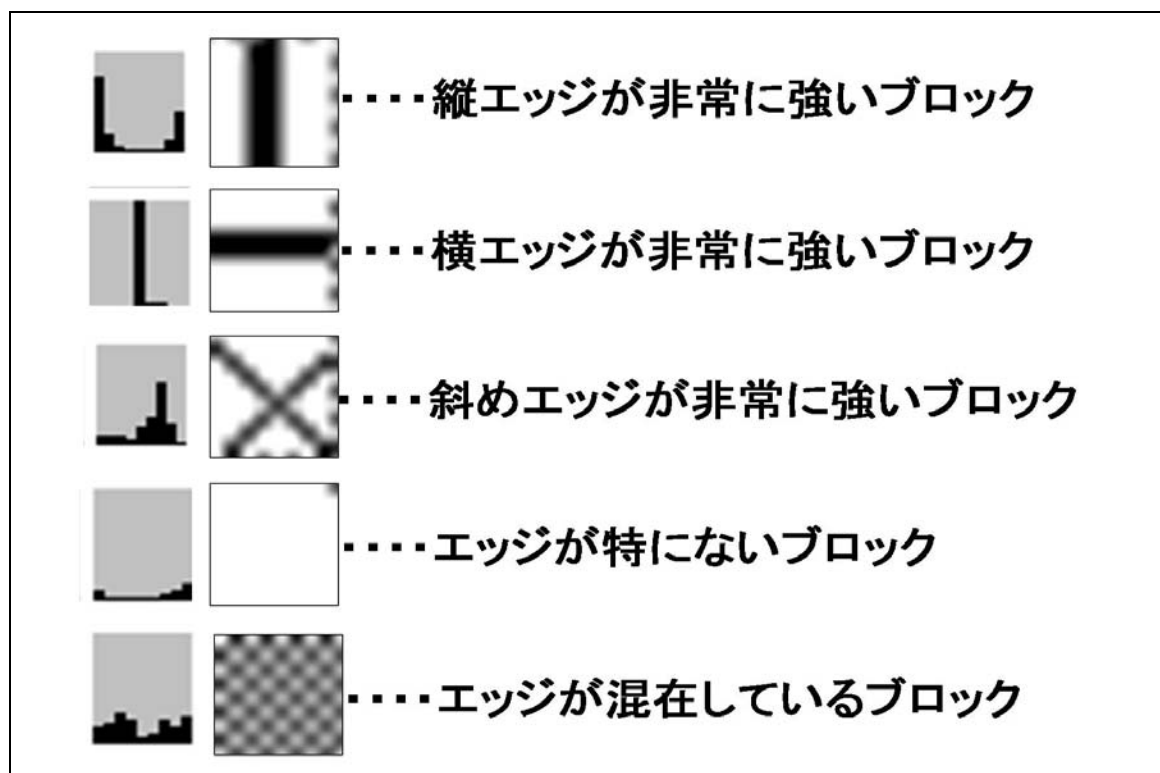


図 4.7 セマンティック HOG 特徴量について

図 4.7 は従来の HOG 特徴量を用いて新たに作成したセマンティック HOG 特徴量の算出手法である．HOG 特徴量は画像をブロック単位に分割し，各ブロック内の輝度情報から量子化されたエッジ方向の強さをベクトル化した特徴量である．図 4.7 中左のヒストグラムが

HOG 特徴量を表す．ヒストグラム中の x 列は量子化されたエッジ方向を表すので，各 x 列の値を見れば，そのブロック内でどの方向にエッジが強いかが判断できる．セマンティック HOG 特徴量はこのブロックがどの方向にエッジが強いかを 5 種類にシンボライズしたものである．図 4.7 中右の画像がセマンティック HOG 特徴量を表す．上からそれぞれ縦エッジが強いブロック，横エッジが強いブロック，斜めエッジが強いブロック，エッジがないブロック，エッジが混在しているブロックである．

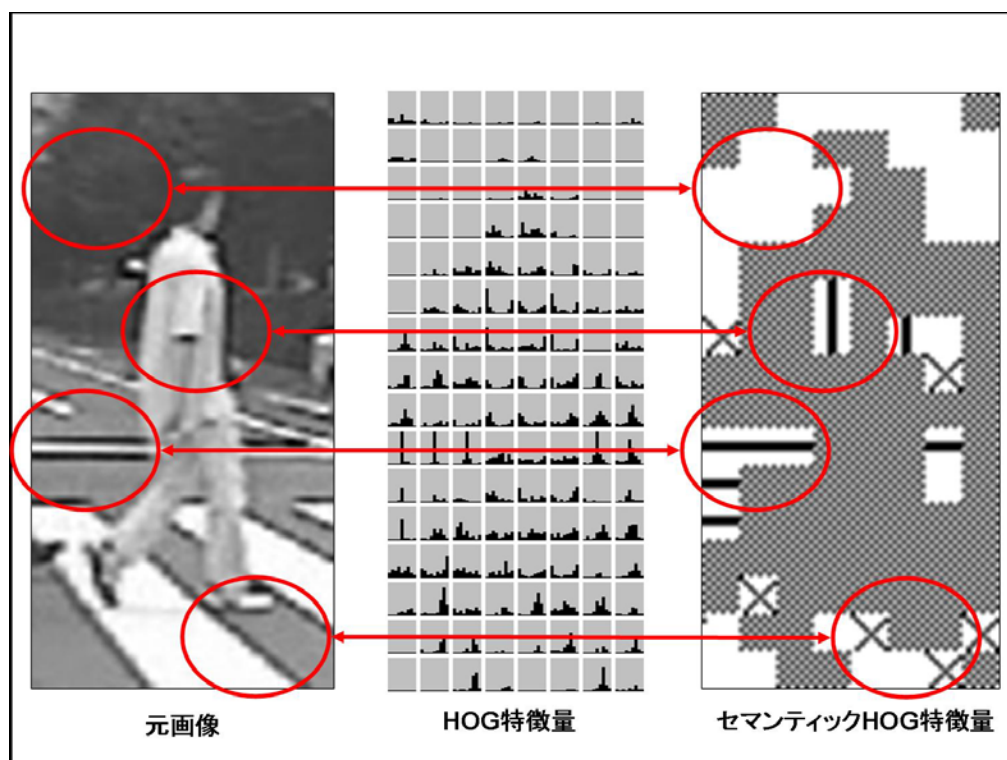


図 4.8 元画像，HOG 特徴量及びセマンティック HOG 特徴量

図 4.8 にある画像に対し HOG 特徴量を抽出した画像とセマンティック HOG 特徴量に変換した画像を示す．元画像左上方の背景の暗い部分はエッジがほとんど立っていない．よってセマンティック HOG 特徴量でもエッジがないブロックとして判断されている．また元画像真ん中と左のそれぞれ人の腕，道路の停止線は縦・横エッジが強いのでそのように判断されている．

以上のようにセマンティック HOG 特徴量を算出した．このようにして算出された特徴量を元画像と比較してみると，歩行者、と非歩行者オブジェクトにはそれぞれお互いに持ち得ない特徴があることが判った．次ページにそれぞれの例を示す．

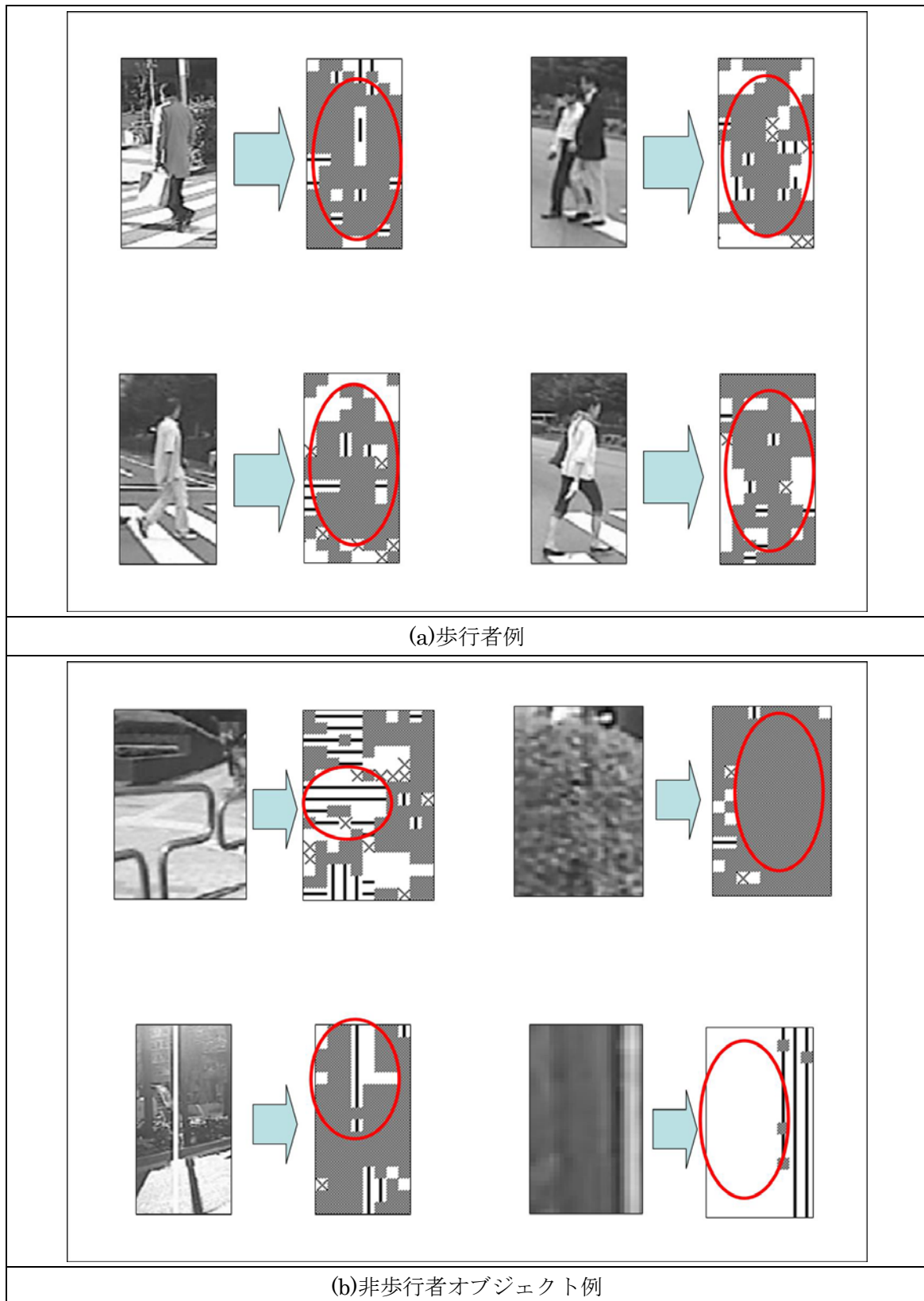


図 4.9 セマンティック HOG 特徴量による歩行者・非歩行者オブジェクト間の差

図 4.9 に歩行者・非歩行者オブジェクトからセマンティック HOG 特徴量を抽出した例について示す. (a)が歩行者の例であるが, 通常ある RO I が歩行者を捉えた場合これらの例のように枠の中心にある「エッジが混在しているブロック」の固まりが見える. これは歩行者自体が服装・姿勢などによりある程度複雑なテクスチャを持つからである. 一方(b)の非歩行者オブジェクトを見るとかなりの特徴があることに気づく. 左上の柵画像は本来歩行者が存在しているはずの中心部位に横エッジが非常に強く発現している. 右上の道路脇の樹木はあまりに一様にエッジが混在している. 左下の太陽光線が画像に写りこんでいる例では画像中心を縦エッジが貫いている. また右下の例ではエッジが画像中にほとんど存在していない. これらのような特徴はいずれも歩行者の例では見られない非歩行者オブジェクト特有の特徴である.

これらの特徴に類する特徴をもつようなオブジェクトを排除する非線形判別器をアルゴリズムに組み込んだ.

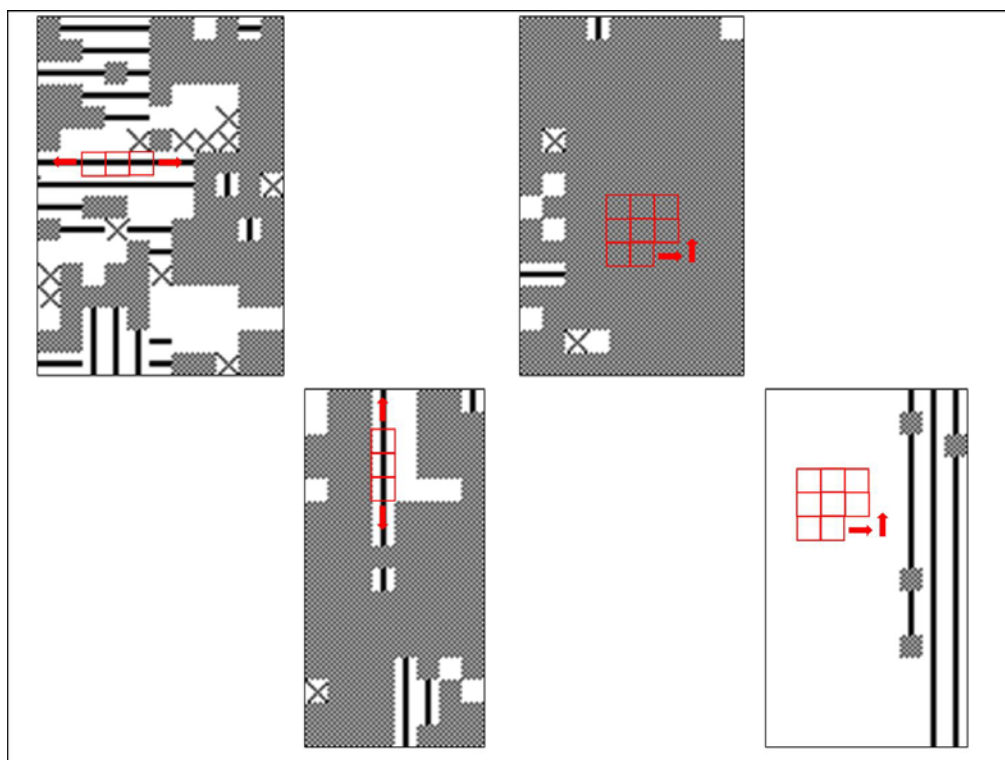


図 4.10 セマンティック HOG 非線形判別器の特徴探索例

図 4.10 は特徴的な非歩行者オブジェクトを除外する非線形判別器の特徴探索例である. 横エッジが強いブロックを発見したら横方向に探索し, 縦エッジが強いブロックが発見されたら縦方向に, エッジが立っていないブロックや混在しているブロックは周囲を探索する. このようにしてそのブロックがどのように分布しているかを判断し, その分布が基準値を超えていたら, HOG/Fisher 判別器による誤報と判断し, この非線形判別器で新たに非歩行

者として判別する．また一様に探索を行うのではなく，歩行者が存在する確率が高いと思われる画像中心に近づくほど値を高く設定するなど工夫も行った．

このように **HOG/Fisher** 判別器の後段に非線形判別器を組みこむことで，誤報率の低下が達成された．後の実験結果の段で結果を述べる．

第5章 歩行者の車両に対する相対軌跡の算出

前章までは、動きベクトル差分によりオブジェクトを検出し、パターン認識によりオブジェクトが歩行者かどうかを判定するという、いわば性能評価の段階であった。本章ではここまでに得た情報を用いて、歩行者 (ROI) の自車両 (カメラ) に対する相対軌跡を導出した。

本研究の最終目的は歩行者との衝突防止システムの実装である。前章までは歩行者検出技術の性能評価を行ってきたが、これだけではシステムの運用上まだ不足がある。ドライバーへの衝突防止の支援を的確に行うためには、車両と衝突しそうな歩行者のみを判断してドライバーに注意を促さなければならない。なぜならば交差点上、特に日本のように比較的狭い道路を歩行者が重なりあって通行しているような状況では、車両付近に何人も歩行者が存在しているのが普通であることや、右左折時には信号待ちの歩行者などは車両に対しかなり接近する事は通常起こり得るからである。そしてこのような歩行者全てに対し注意を促すことは、交差点を曲がるたびにシステムによる発報が行われ、ドライバーのシステムへの受容性に大きな影響を与えることになる。このようにただ車両付近に歩行者がいるというだけで発報を行うことは非現実的であり、真に危険な状態にある歩行者のみに対し発報が行われる必要がある。

本研究では、発報の必要な歩行者を検出された歩行者の中から選別するために、歩行者の自車両に対する相対軌跡を用いた。検出された歩行者全てに対し相対軌跡を導出し、高い衝突可能性を持つ歩行者にのみ発報をすれば、上記の問題は解決され、実システムへの運用も可能になる。第4章における幾何拘束において非歩行者 ROI を除外するために、本アルゴリズムはカメラから ROI までの距離と、カメラの中心軸からの角度を得た。ROI の下辺中心を歩行者の足元と仮定すると、カメラからの距離と角度より対象 ROI の相対軌跡が得られる。

さて、以上の手法により相対軌跡が導出できるが、車両の加減速や路面の凹凸などによるカメラの振動、また ROI 下辺が歩行者の足元と正確に一致しない場合も考えられる。そこで得られた相対軌跡が正しい軌跡を描いているかの検証を行った。

検証を行うために、2台の同期カメラを用いた。図 5.1, 図 5.2 に検証画像を一連の流れにそって示す。それぞれ直進シーンと右折シーンに対し道路脇に立っている歩行者の登場から抜けるフレームまで示してある。画像中上列がそれぞれ左カメラ・右カメラによる映像である。また下列左が車載カメラによる撮影画像であり、相対軌跡はこれをもとに導出される。そして下列右がそのフレームまでの相対軌跡である。

以下に検証手法を述べる。まず広い平面上にグリッドを構成する、そして各交差点上に車両を配置しその様子を同期カメラで撮影し、キャリブレーションを行う。その後あらかじめ設定された点上に歩行者を配置し、コース上を車両を走行させ撮影した映像に対し処理

をかけ、歩行者の ROI 情報を得て相対軌跡を導出する。そして車両走行の様子は同期カメラで撮影しており、また歩行者位置も既知であるため、三角測量により実際の絶対軌跡を導出する事ができる。

図 5.3 は上記の手法により算出された相対軌跡、絶対軌跡を示す。図上青点がシステムにより導出された相対軌跡、赤点が三角測量により得られた絶対軌跡、また赤線があらかじめ設定されたコース上を正確に走行した場合の正解データである。

実験は直線走行時と曲線走行時の 2 通りを行った。図 5.1 が直線走行時、図 5.2 が曲線走行時をしめしている。検証結果は両シーンのシステムにより相対軌跡と、三角測量による絶対軌跡の平均の誤差は約 30cm で、カメラ振動などによる影響はそこまで大きくなく、実システムへの適用が充分可能であるという結論を得た。

図 5.4(a)はあるシーンに対する本研究の検出結果例と、(b)はそのシーンに登場するオブジェクトごとの相対軌跡である。オブジェクト番号 18 の軌跡は第 6 章で説明する ROI 修正アルゴリズムを適用してあるので二通りの軌跡を示してある。赤い軌跡が適用前の相対軌跡、青い軌跡が適用後の相対軌跡である。

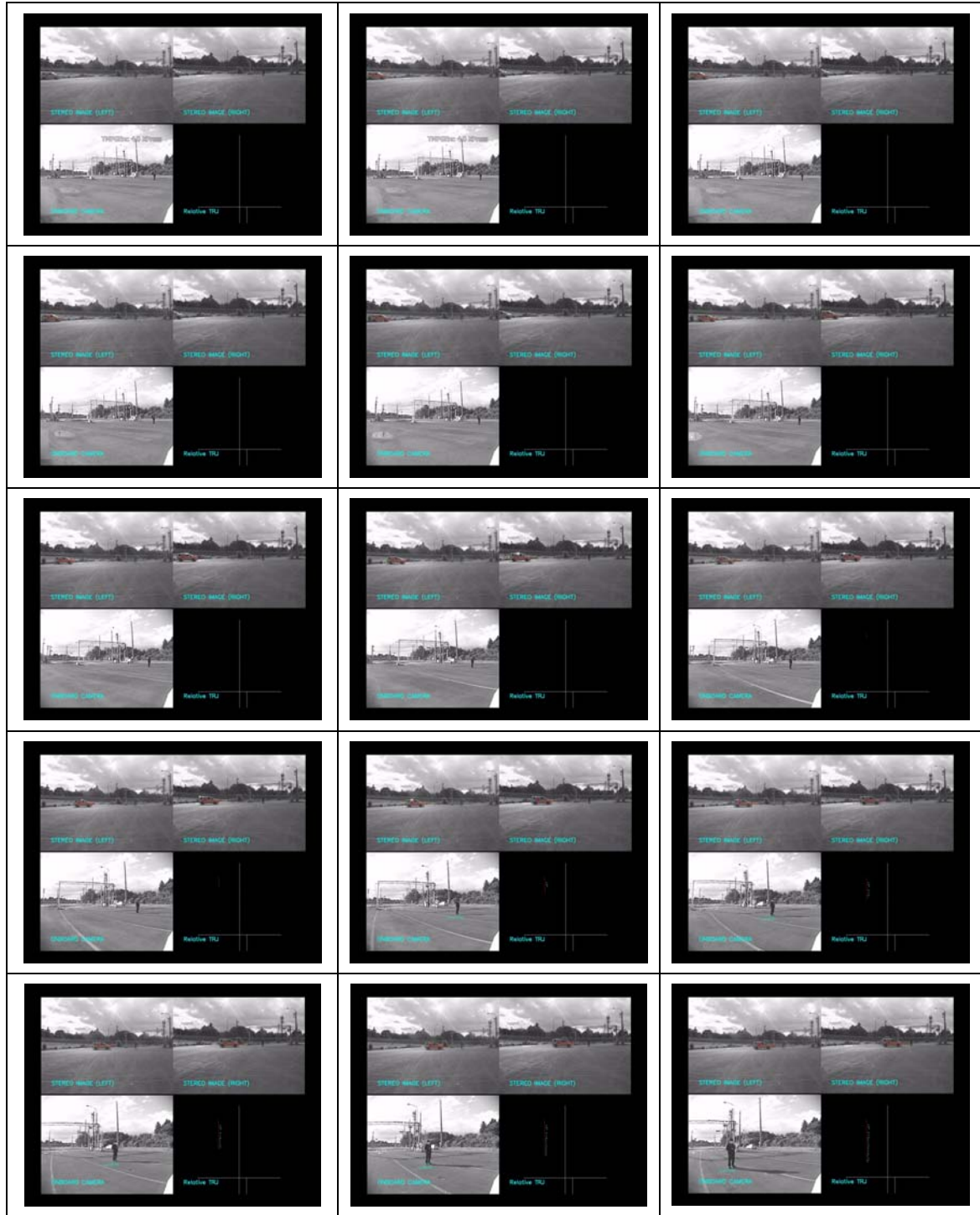


図 5.1 相対軌跡検証実験 直進シーン

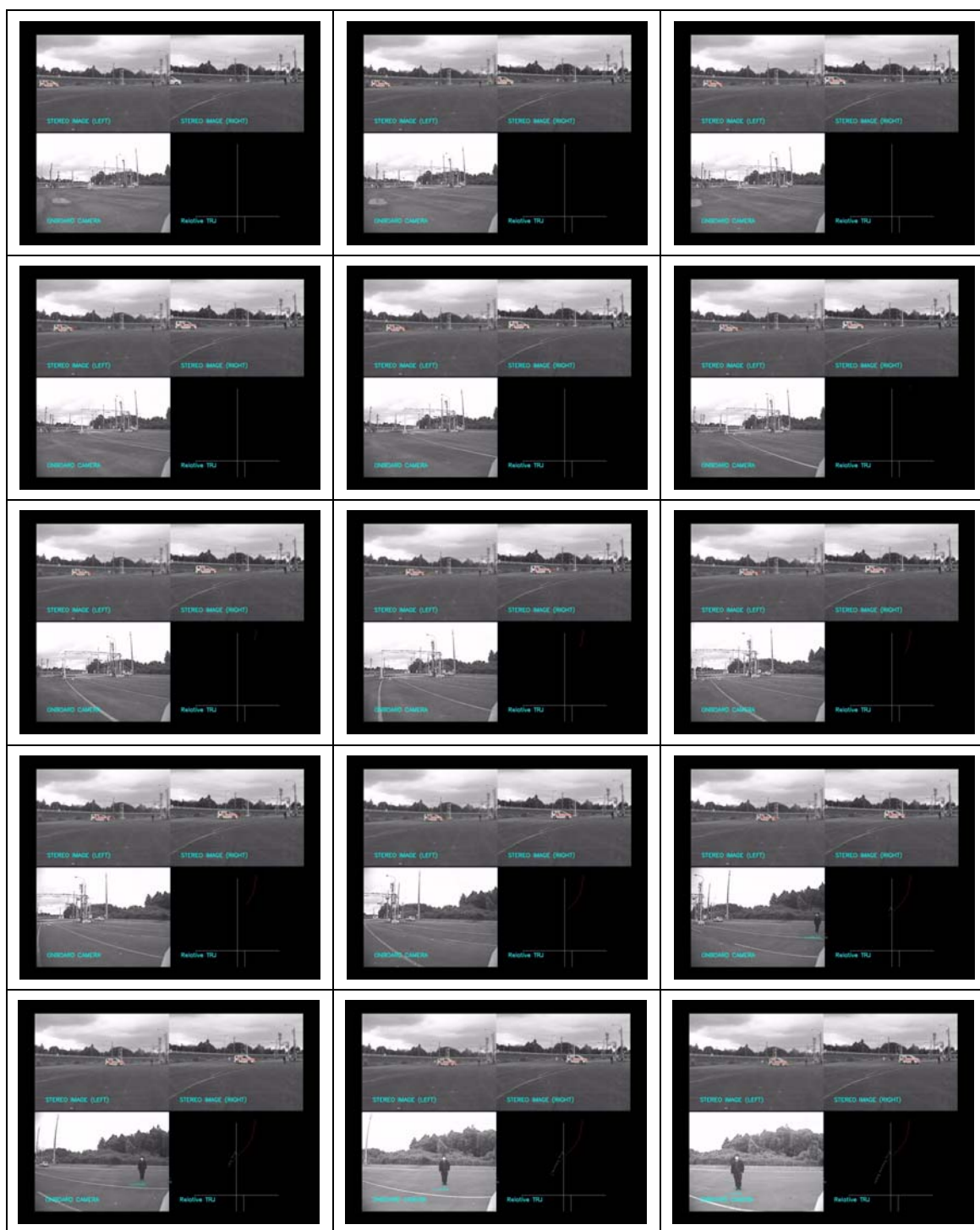


図 5.2 相対軌跡検証実験 右折シーン

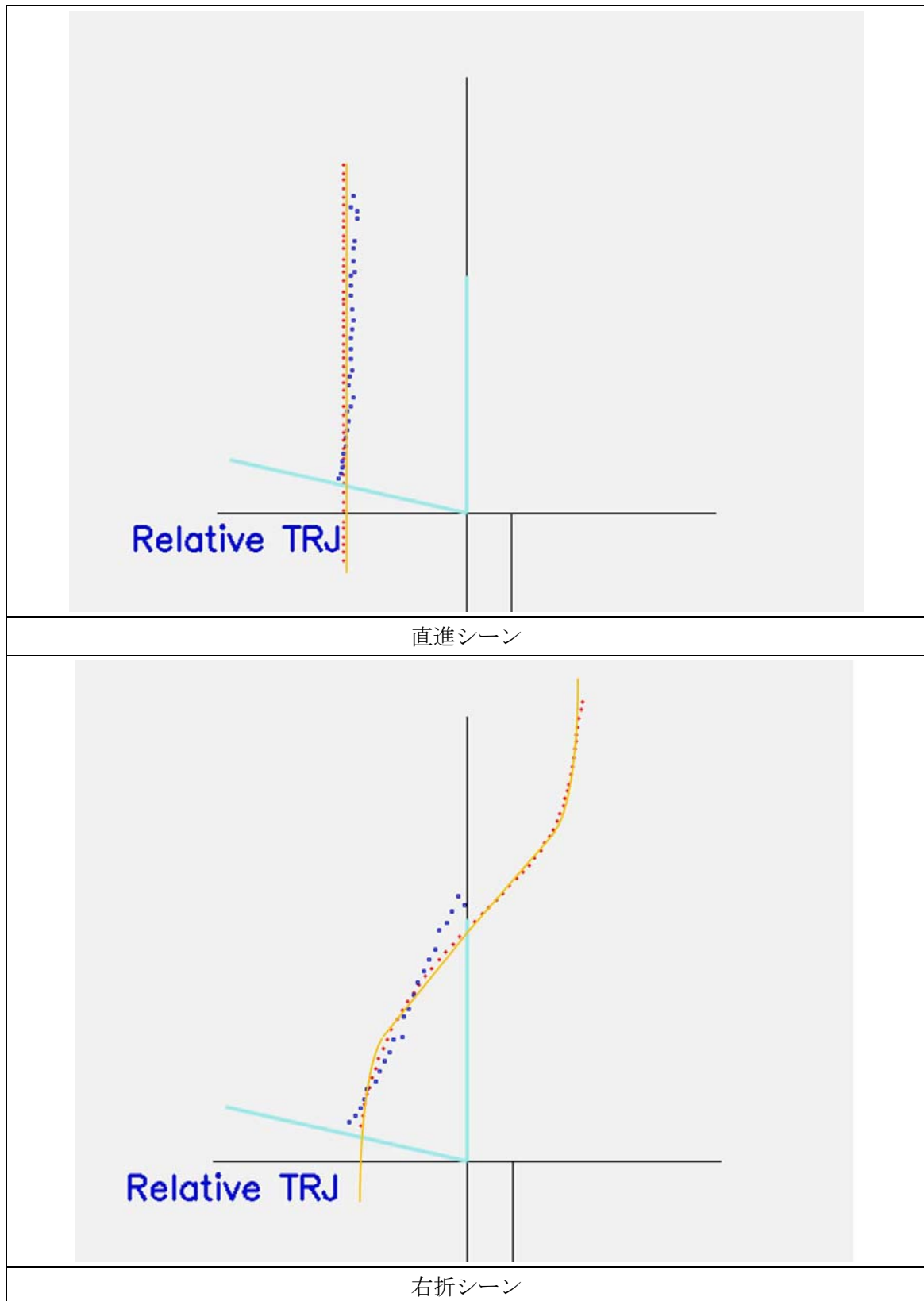
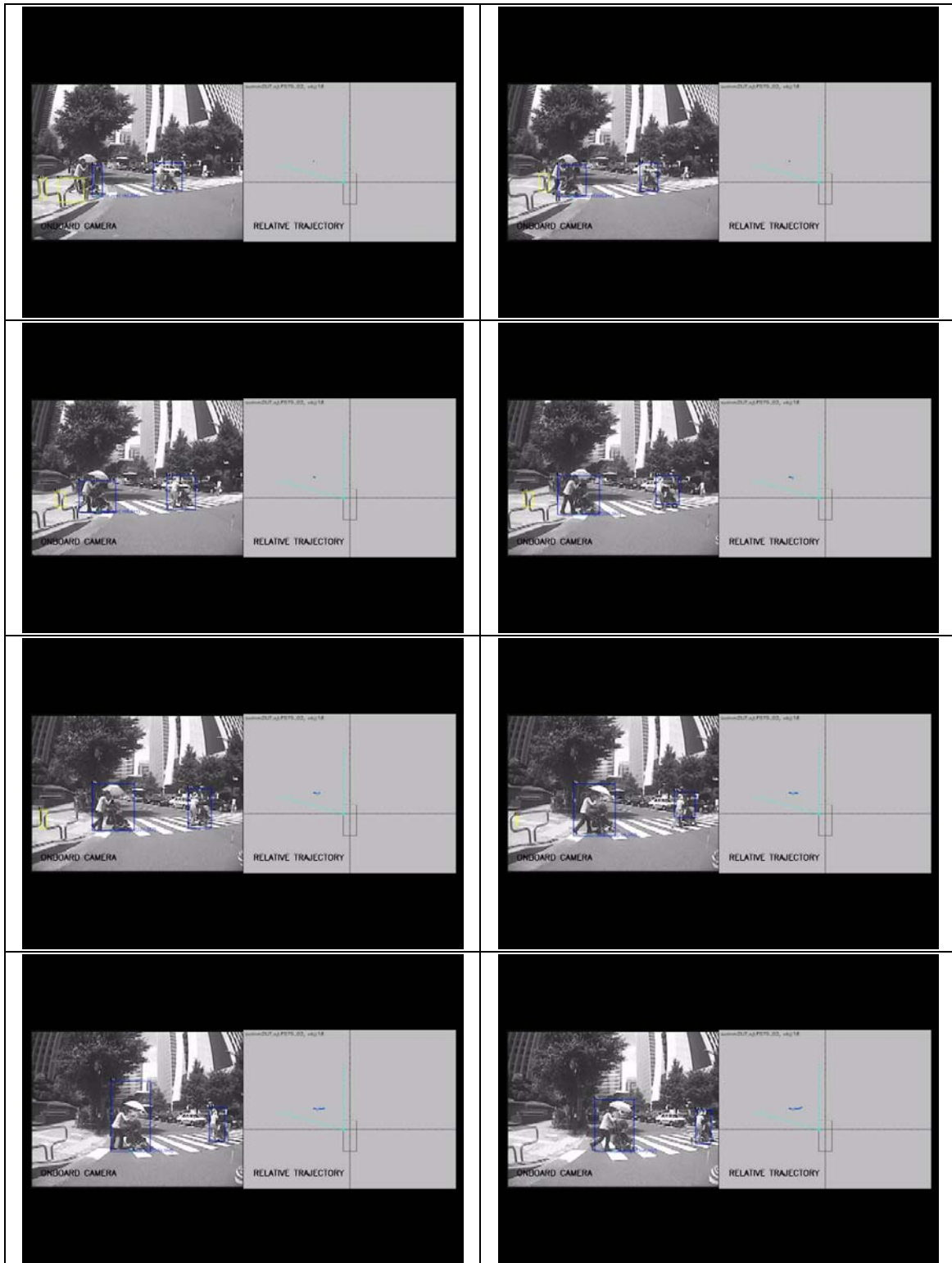


図 5.3 正解データ(赤線)・絶対軌跡(赤点)・相対軌跡(青点)



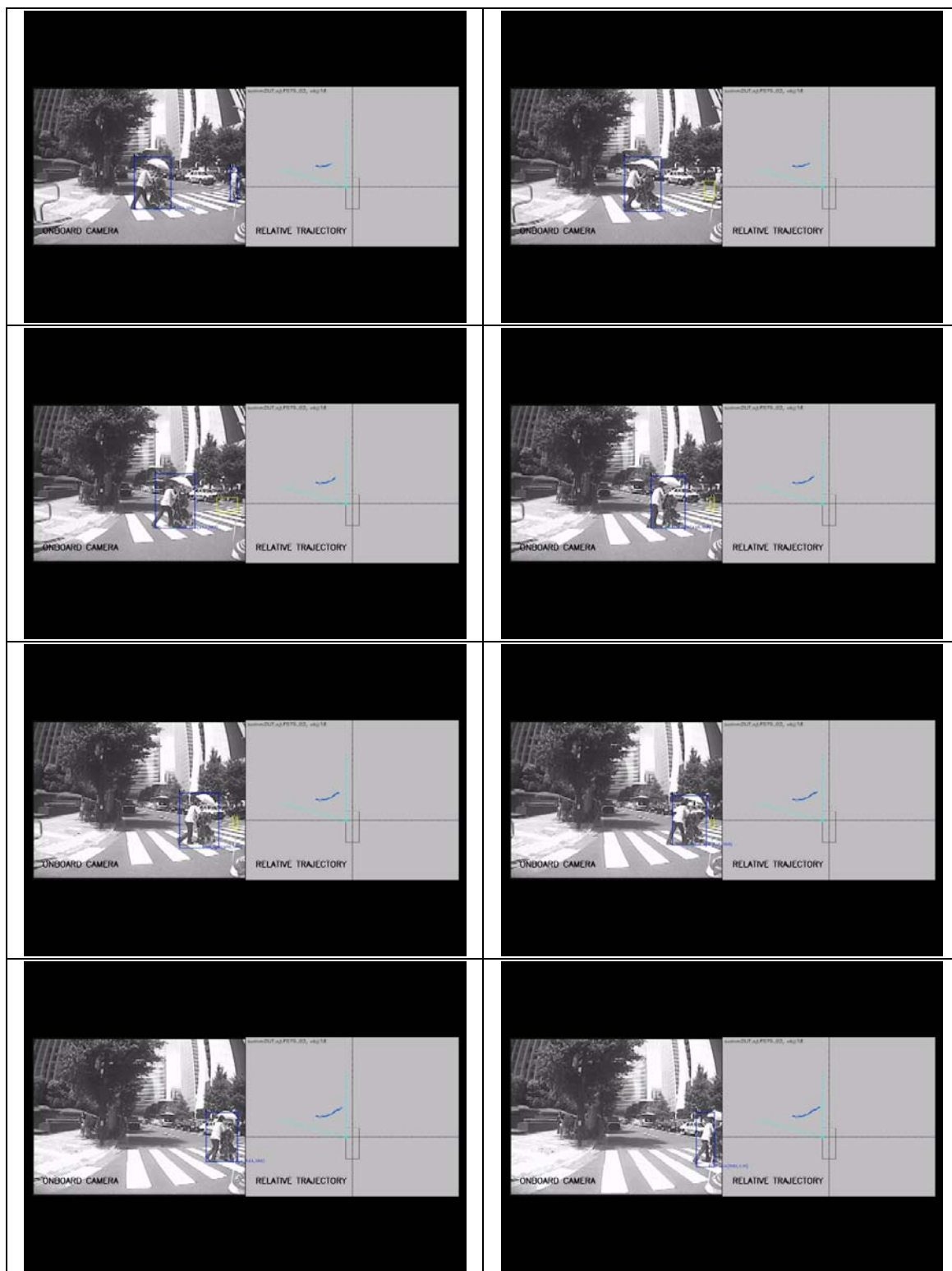
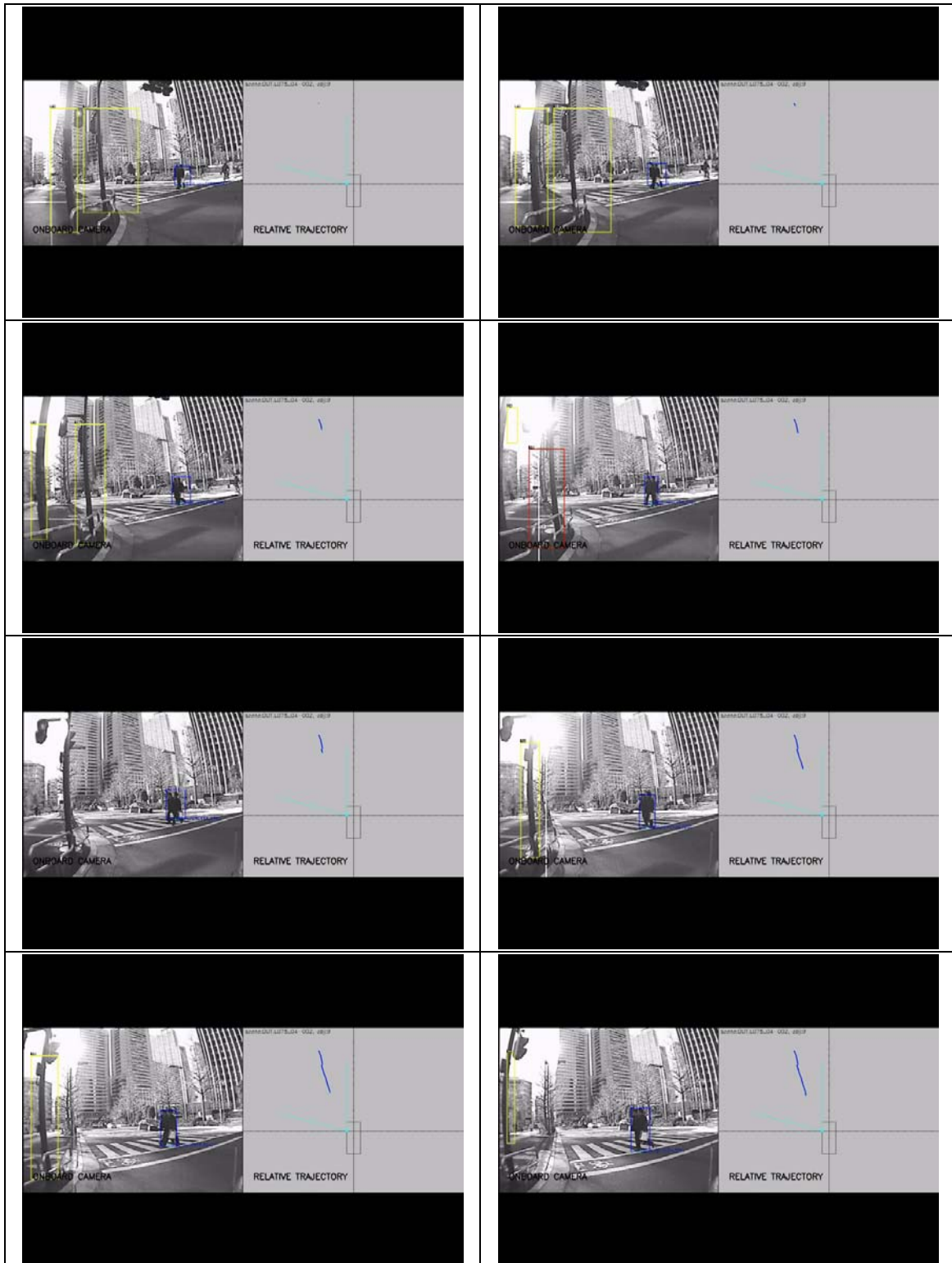


図 5.4.1 相対軌跡取得例 ケース 1



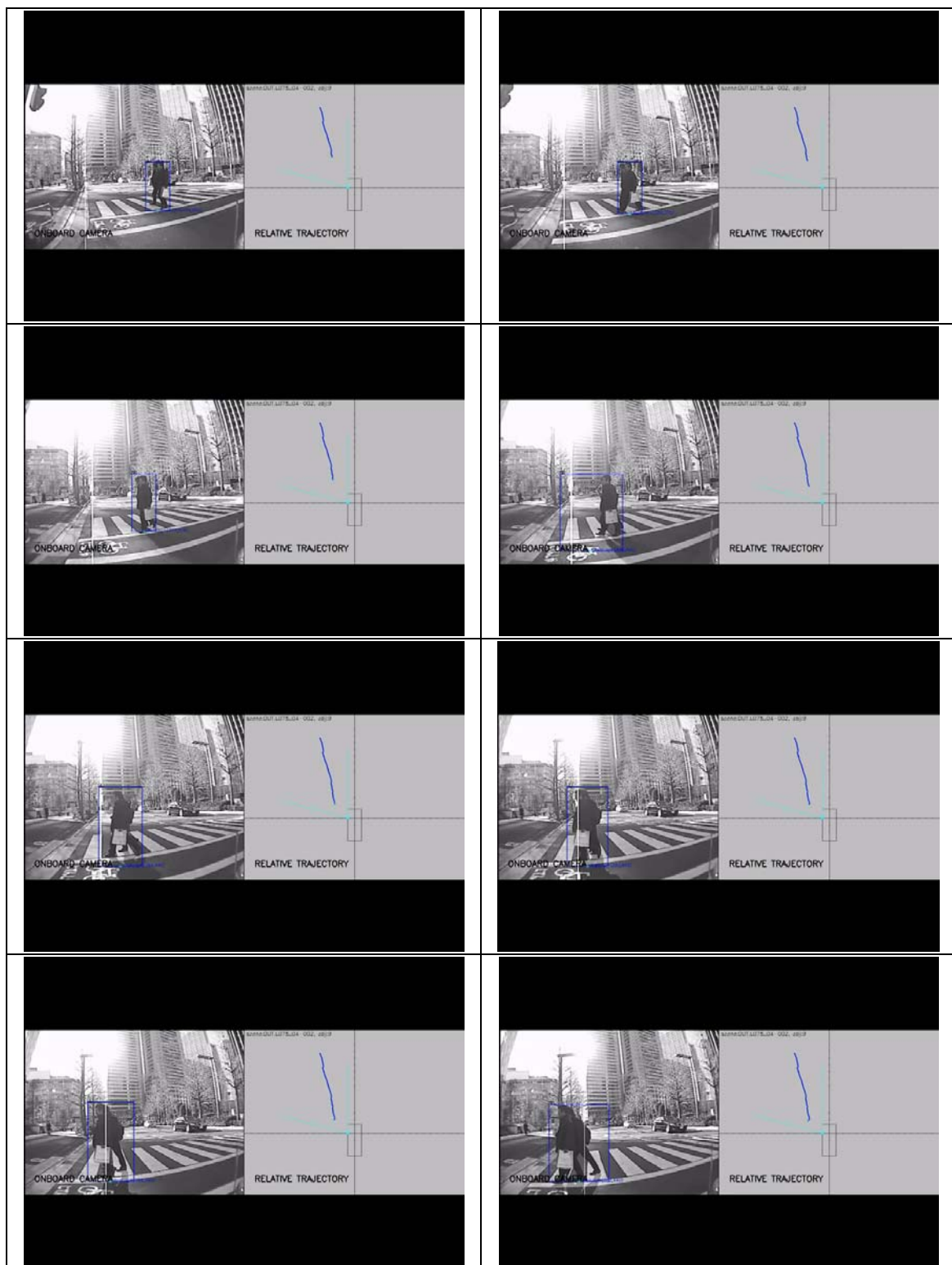


図 5.4.2 相対軌跡取得具体例 ケース 2

第6章 ROI 情報修正アルゴリズム

オブジェクト検出の段では動き差分情報を参照して，ROI を検出する．しかし何らかのエラーにより ROI が正確にオブジェクトを捉えていないためケースが存在する．このような状況は，歩行者検出の精度を低下させ，また前章で述べた相対軌跡の正確性にも影響する．本章では相対軌跡を正確に算出するための ROI 情報の修正アルゴリズムを説明する．

6.1 オブジェクトの ROI 情報修正

時空間 MRF モデルは動きベクトルの差異を元に、領域を決定し ROI として抽出する。そしてパターン認識では ROI が正確に算出できないケースを想定し、抽出した ROI 内を複数のサイズで走査する事を行っているので、多少 ROI が正確に歩行者を捉えていずとも HOG/Fisher による判別で誤判定にはなりにくい。しかし本研究は実システムへの適用を目標にしているため、相対軌跡の導出においては ROI が正確に算出できない事は相対軌跡を歪ませる原因になるので、実システム上大きな問題になりうる。そこで過去の ROI 情報を参考に現在の ROI 情報を修正するアルゴリズムを提案・実装した。

6.1.1 ROI 拡張アルゴリズム

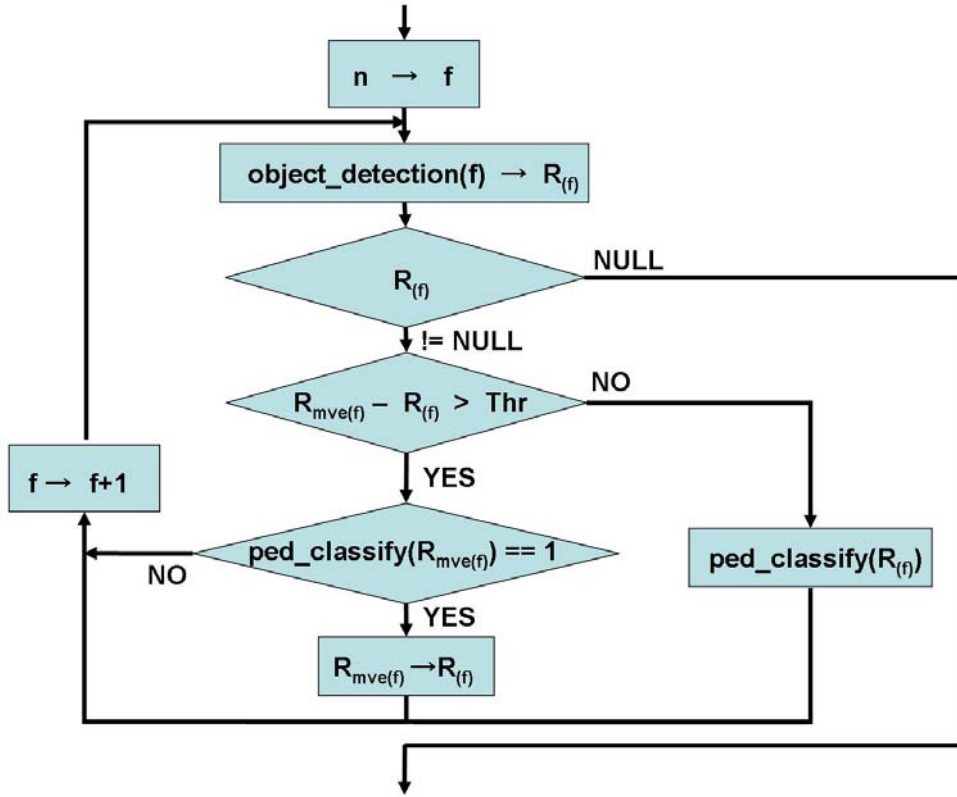


図 6.1 ROI 情報拡張アルゴリズム

[定義]

{定数}

n : 処理を始めるフレーム番号

Thr_{ext} : ROI 情報 R_{θ} の拡張の際の閾値

{変数}

f : 処理中のフレーム番号

R_{θ} : “ f ”フレームにおける検出オブジェクトの ROI 情報

$R_{mve(\theta)}$: “ n ”フレームから “ f ”フレームまでの時系列情報を元にカルマンフィルタを用いて推定された R_{θ} の最小分散推定値

$M(R_{mve(\theta)})$: 時空間 MRF による $R_{mve(\theta)}$ が示す領域における動き差分量

{関数}

$object_detection(f)$: フレーム “ f ” 中に存在するオブジェクトを検出し,存在するならば対象オブジェクトの ROI 情報 R_{θ} を返す

$ped_classify(R_{\theta})$: “ R_{θ} ” で表される領域が歩行者か非歩行者かを HOG 特徴量を用いて判別する. もし歩行者なら 1 を,そうでないなら 0 を返す.

オブジェクト本来の形状と検出された ROI 情報の比較検証を行ったところ、ROI 情報がオブジェクト本来の形状を正確に捉えていないケースが存在することがわかった。時空間 MRF モデルは画像をブロックに分割し、ブロックマッチングにより動きベクトルを計算することによりオブジェクトを検出しているが様々な要因により検出が正常に行われない事がある。

まず本項で扱うのは本来の形状より欠けて検出されているケースである。例えば道路上で歩行者は基本的に規則的に歩行するので、歩行者の脚部の規則的な前後運動から動きベクトルが乱され脚部の検出に失敗する、などのケースが主に挙げられる。上述のようなケースでは ROI 情報が歩行者の脚部を正確に検出していないので、判別器に入力される画像が歩行者の上半身だけの画像になり、歩行者として判別できなくなる。また前章で述べた相対軌跡についても、相対軌跡の算出には ROI の下辺情報の中心位置を足元として用いているため、下半身が欠損した ROI 情報では軌跡が歪んでしまい正確な軌跡が算出できず、軌跡による危険歩行者の判別精度が低下することとなる。このような ROI 下辺情報の欠損を修正するためにカルマンフィルタを用いた図 6.1 で表される ROI 情報の拡張アルゴリズムを用いた。

前頁図 6.1 下に図 6.1 中に用いた変数・関数などを記載した。以下にアルゴリズムの詳細を示す。まず f は処理中のフレームを表し、ここでは ROI 下辺情報の修正が n フレームから開始されると仮定する。アルゴリズムの最初の処理は時空間 MRF モデルが f フレーム目の画像中に含まれるオブジェクトを検出する事である。そしてその検出されたオブジェクトの ROI 情報が R_{θ} の形で返される。もし f フレーム中にオブジェクトが存在しない場合には R_{θ} に NULL が返され処理は終了する。次に、オブジェクトが検出された場合アルゴリズムは $R_{mve(f)}$ と R_{θ} の差を算出する。 $R_{mve(f)}$ は処理開始の n フレームから現在処理中の f フレームまでのカルマンフィルタによる R_{θ} の最小分散推定値である。この差というのは、要は ROI 情報が最小分散推定値と比較して小さい方向にある場合には ROI 情報拡張へ、大きい方向にある場合には後に述べる ROI 情報分割へとシフトさせるスイッチである。そして差が閾値 Thr_ext より大きい場合には $R_{mve(f)}$ で示される領域がパターン識別による歩行者識別器へと入力される。判別の結果が 1 である場合（ $R_{mve(f)}$ で示される領域が歩行者であると判断された場合）には R_{θ} を $R_{mve(f)}$ に更新して次フレームに処理を移る。また差が Thr_ext より低い場合には R_{θ} の更新は行われずそのまま次フレームに移行する。以上が ROI 情報拡張アルゴリズムの一連の流れである。図 6.1 では単一のオブジェクトのみ存在すると仮定して処理アルゴリズムを描写しているが、通常 1 フレーム中には複数のオブジェクトが存在する。その場合についても基本的な処理に変更はない。

ROI 情報を修正する際、ROI 情報の時系列データを用いて行う画像空間上での修正も、実際に相対軌跡を導出してからの実空間上での修正も意味的に等価な作業を行っていることになるが、本研究では実空間上の修正にはその分余計な計算コストがかかると判断し、画像空間上での修正を行っている。

6.1.2 ROI 分割アルゴリズム

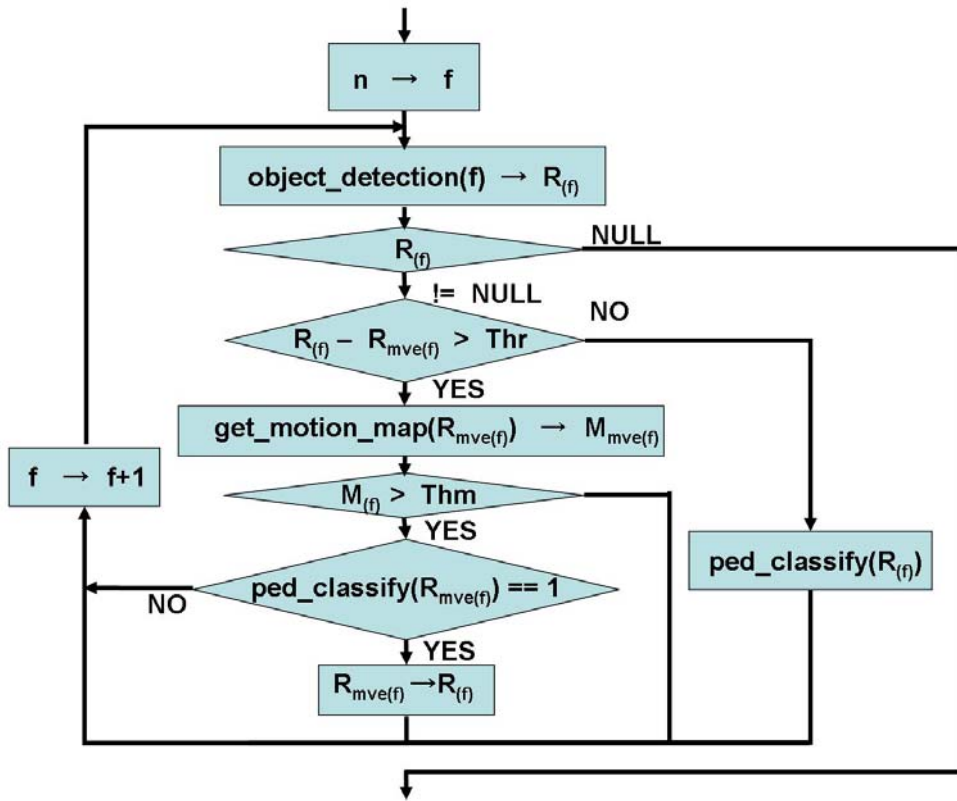


図 6.2 ROI 情報分割アルゴリズム

[定義]

{定数}

Thr_seg : ROI 情報 $R(t)$ 分割の際の閾値

Thm_seg : ROI 情報 $R(t)$ 分割の際の $M(R_{mve(t)})$ の閾値

{変数}

$M(R_{mve(t)})$: 時空間 MRF によって算出された $R_{mve(t)}$ の動き差分情報

{関数}

$get_motion_map(R(t))$: $R(t)$ で表される領域の動き差分情報マップを取得する

※ その他定数・変数・関数については 6.1.1 項で記述したものの同様である。

6.1.1 項では ROI 情報が欠損する場合の拡張アルゴリズムについて述べたが、ROI 情報にエラーが生じる場合はもう 1 種類存在する。それは ROI 情報が現実のオブジェクトの形状より拡大して捉えてしまう場合である。時空間 MRF モデルは図 3.3 で表されるように動き差分が存在するブロックで構成される領域を 1 オブジェクトとして検出する。しかし交差点での右左折時など、例えば歩行者の近辺にポールや高いビルが存在している場合など

は、動き差分が混在しているので ROI 情報の上辺が背の高いポールやビルに引っ張られてそれらも含んで一つのオブジェクトとして検出を行ってしまう。そしてこの様に歩行者以外のオブジェクトを多く含んだ ROI 情報はパターン判別器でも非歩行者と判定されることが多く、また四章で示す幾何拘束においても拘束を受けやすい。また通常このようなケースは複数フレームに渡って続くので欠報率の大幅な低下も問題になる。このような場合の ROI 情報を修正するためには 6.1.1 で示す拡張アルゴリズムでは対応できないため、ROI 情報分割アルゴリズムを実装した。

アルゴリズムの詳細を図 6.2 に示した。用いている変数で 6.1.1 と異なるものには新たに説明を付加してある。アルゴリズムの R_{ℓ} の算出までは 6.1.1 の ROI 拡張アルゴリズムと同様である。次に R_{ℓ} と $R_{mve(\ell)}$ の差をとりその差が閾値 “ Thr_seg ” を越えた場合についてのみ次の処理に進む。本項は分割アルゴリズムであるので R_{ℓ} 中には十分な動き差分が存在すると考えられる。よって $R_{mve(\ell)}$ で示される領域中に人として考えられるレベルの動き差分が存在するかをチェックする。 $R_{mve(\ell)}$ 中の動き差分を “ $M_{mve(\ell)}$ ” 代入し、その値が閾値 “ Thm_seg ” を越えた場合に $R_{mve(\ell)}$ パターン識別に inputs する。

6.1.1 の拡張アルゴリズムと本項の分割アルゴリズムの適用結果について次項で示す。

6.2 ROI 修正アルゴリズム適用結果

前項で、何らかのエラーが原因で ROI 情報に異常が起こる際の修正アルゴリズムの詳細について説明した。本項ではそれらの修正アルゴリズムを適用し、ROI 情報に修正が施された結果について示す。

図 6.3.1～6.3.4 には 6.1.1 で示した ROI 拡張アルゴリズムの適用例について示す。画像中拡張アルゴリズムが適用される ROI には ROI の下辺に印を付加してある。画像中赤枠が拡張アルゴリズム適用前を示し、適用後もしくは適用なしのフレームは青枠で示される。どの適用例も適用前は足元が欠けているのがわかる。また適用画像の下に第 5 章の手法により算出された相対軌跡を示してある。ROI の下辺情報が正確にとれるようになったことから、欠報数を低下させると共に相対軌跡がより正確に導出できるようになっているのがわかる。

図 6.4.1～4 には 6.1.2 で示した ROI 分割アルゴリズムの適用例について示す。それぞれ上段に分割アルゴリズム適用前、下段に分割アルゴリズム適用後の結果を示してある。一段に入りきらない場合は二段に分けて示している。青枠が歩行者と判定された ROI、赤枠・黄枠が非歩行者と判定された ROI、黒枠が判定準備段階の ROI である。図 6.4.1 では一瞬背景のビルと同一オブジェクトとして検出されているため欠報になっているが、アルゴリズムを適用していた結果 ROI が分割され、歩行者と判定されていることがわかる。また図 6.4.2 では交差点横断歩道前に立っている男性二人組みが後ろの高いポールに引っ張られて欠報となっているが、適用後はポールと男性二人組みがしっかりと分割されている。同様のケースは図 6.4.3 や図 6.4.4 でも見られる。と歩行者近辺に存在するオブジェクトと同一オブジェクトとみなされて欠報になっていた例が、再分割を行うことにより歩行者と判定されていることがわかる。

このようにただ時空間 MRF モデルから出力される ROI 情報を用いてパターン認識を行うのではなく、現フレームまでの情報を鑑みて最も確からしい ROI 情報を時空間 MRF モデル側にフィードバックする処理により更なる精度の向上を達成した。

これらの手法を適用した結果については第 7 章で論ずる。

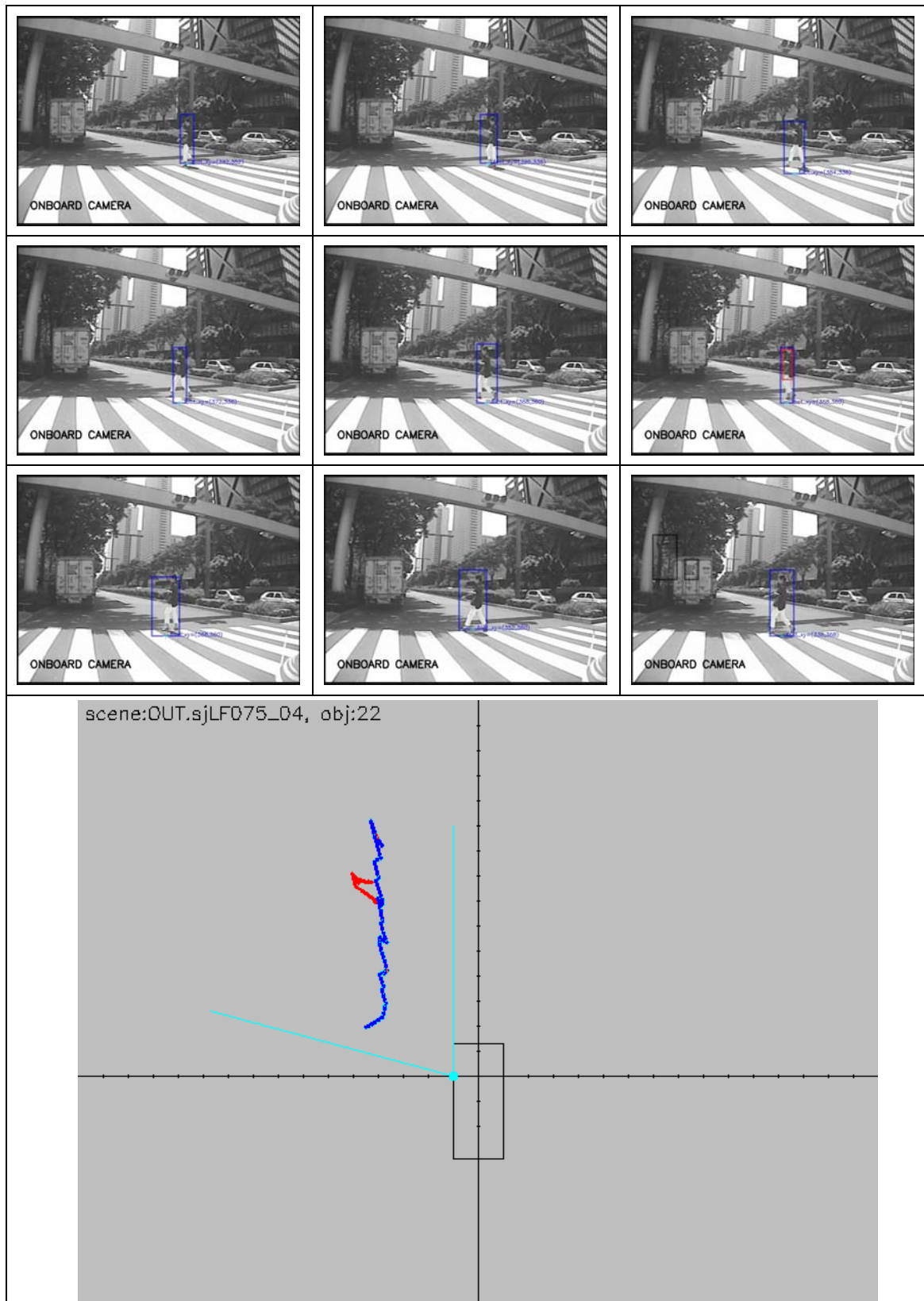


図 6.3.1 ROI 拡張アルゴリズム適用例 ケース 1

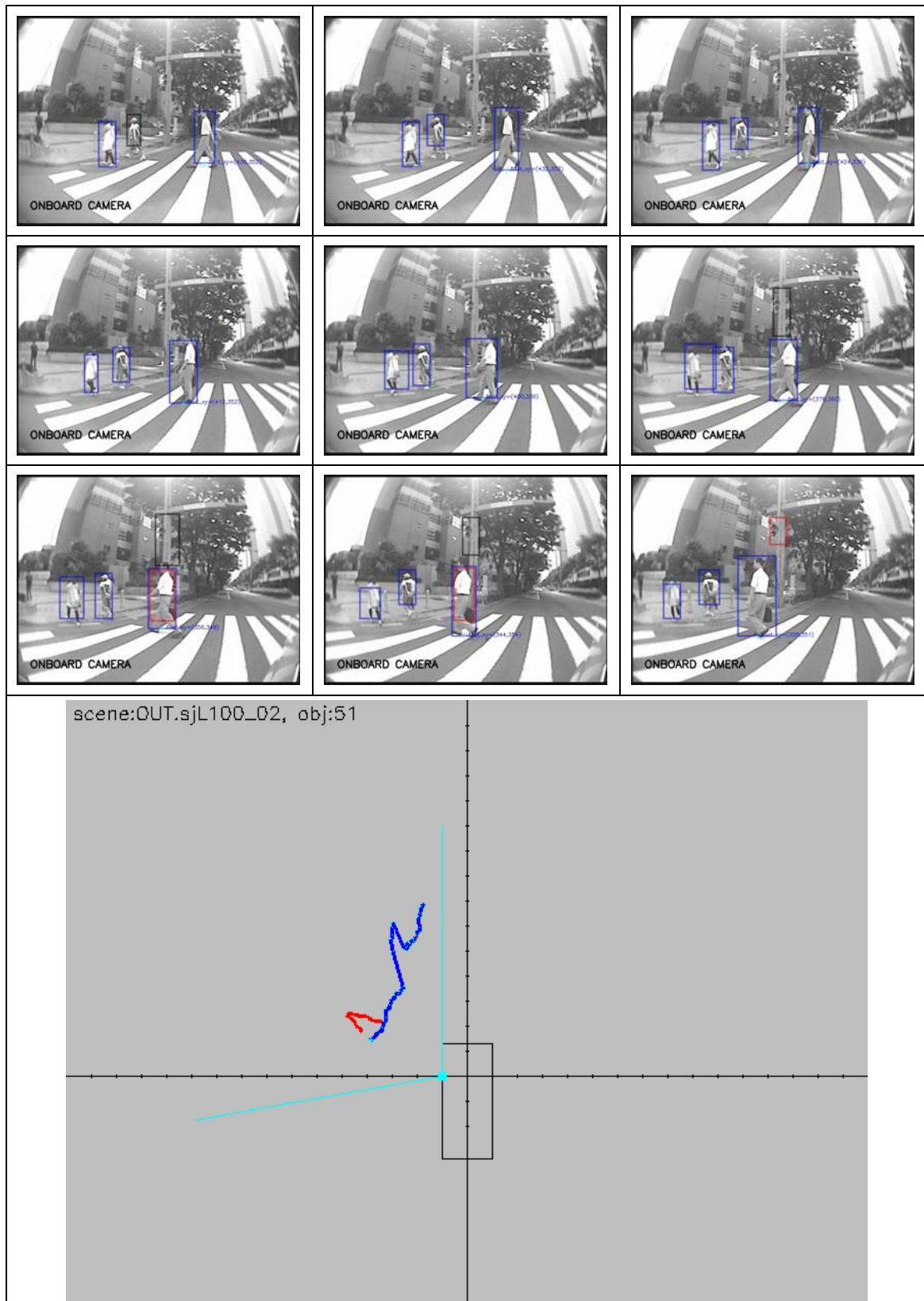
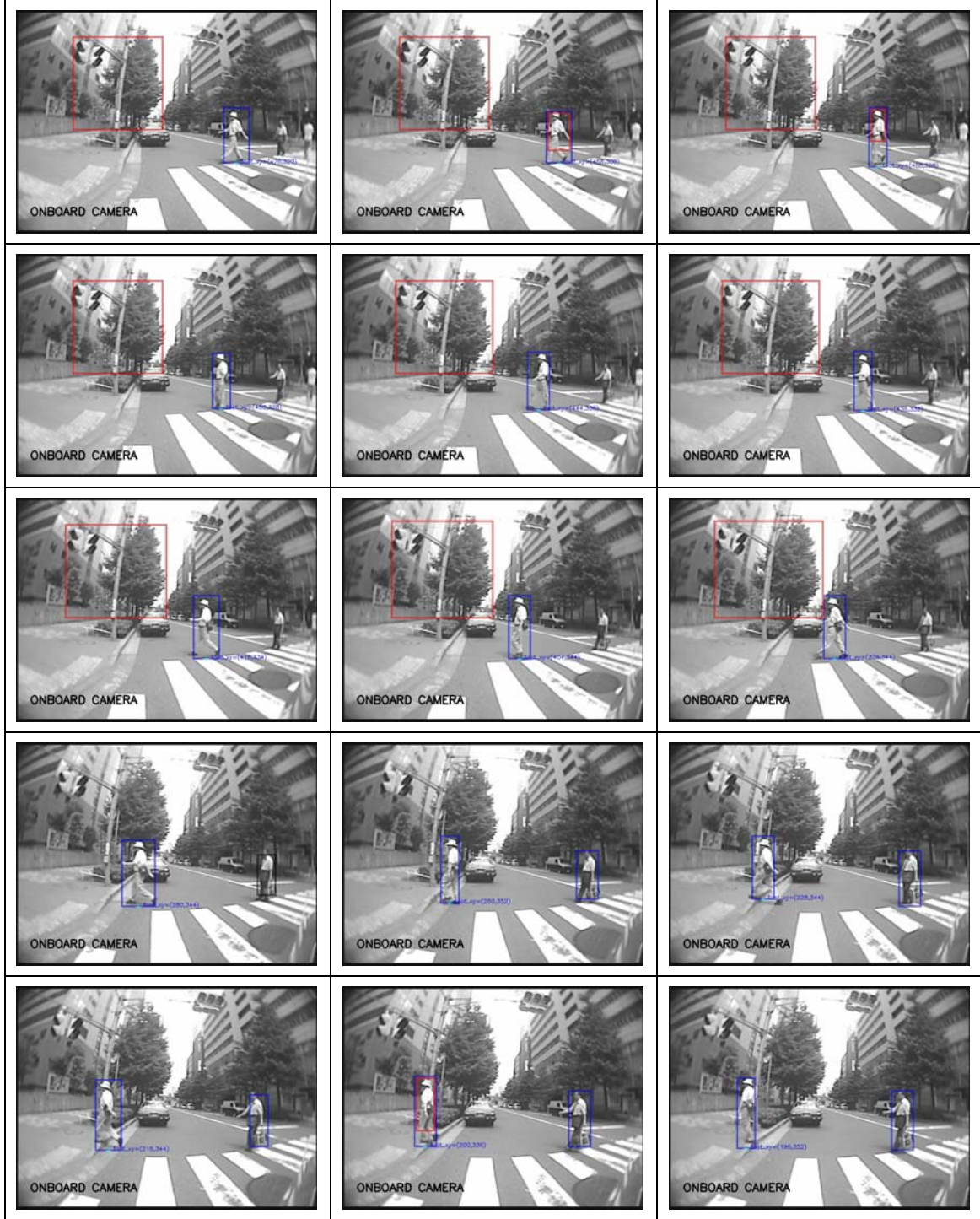


図 6.3.2 ROI 拡張アルゴリズム適用例 ケース 2



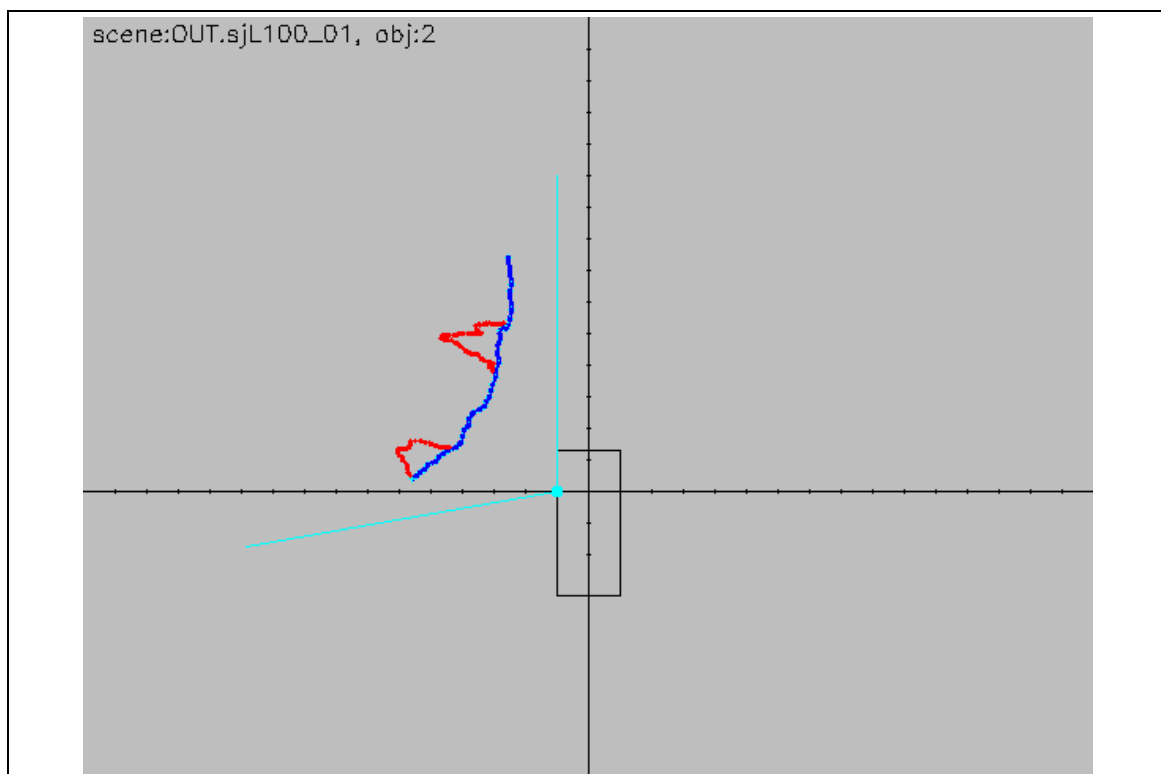
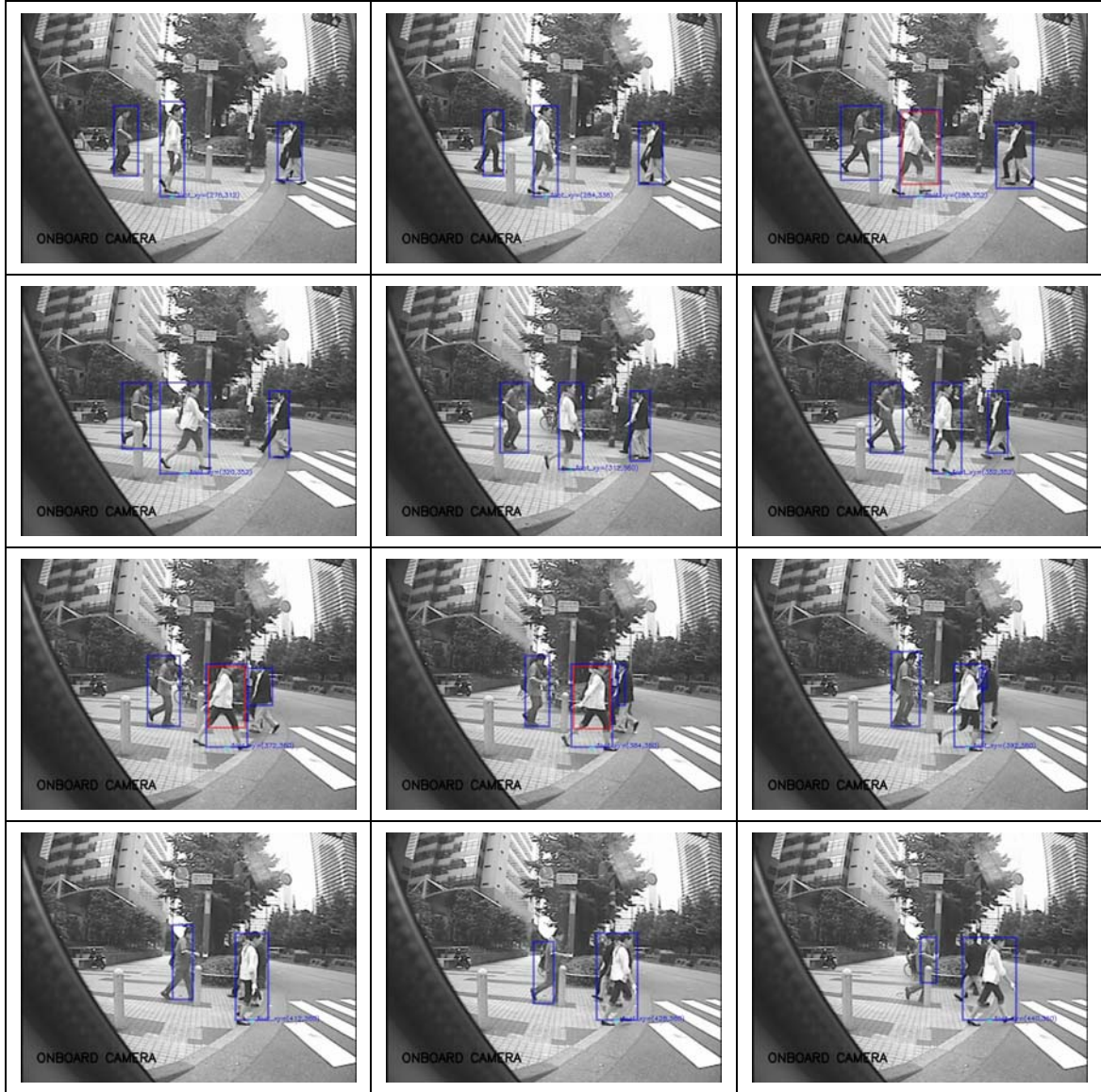


図 6.3.3 ROI 拡張アルゴリズム適用例 ケース 3



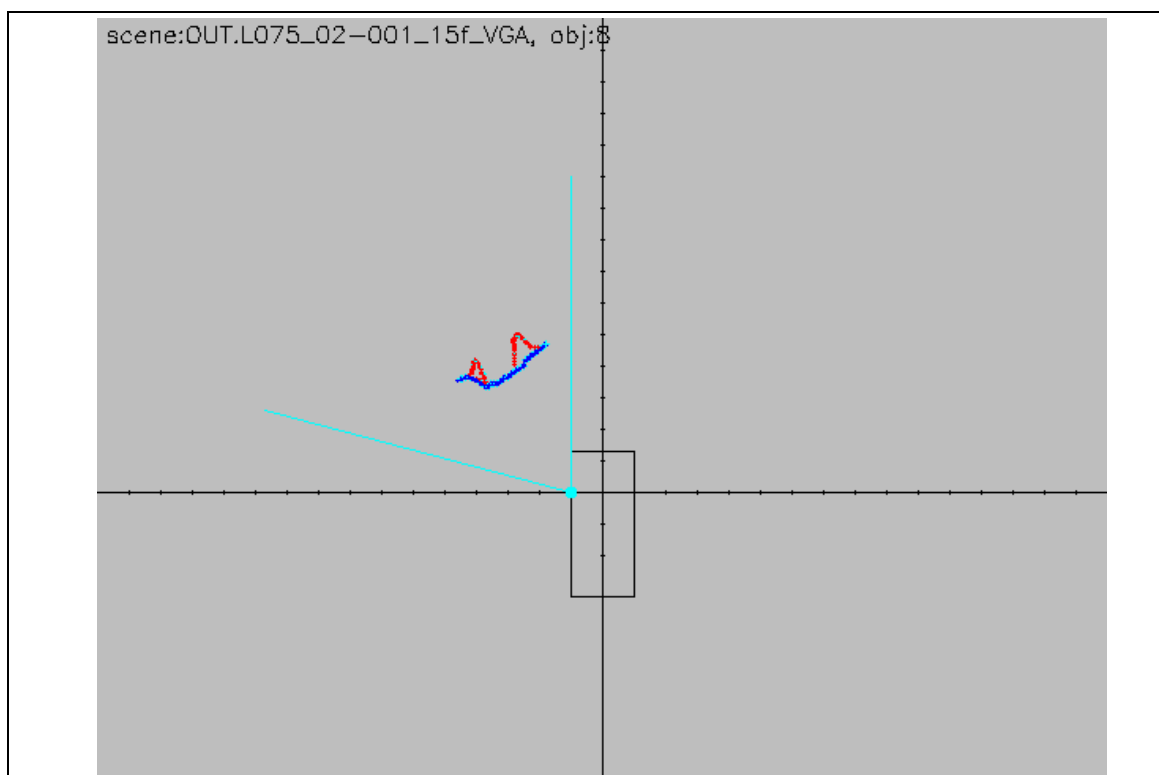


図 6.3.4 ROI 拡張アルゴリズム適用例 ケース 4

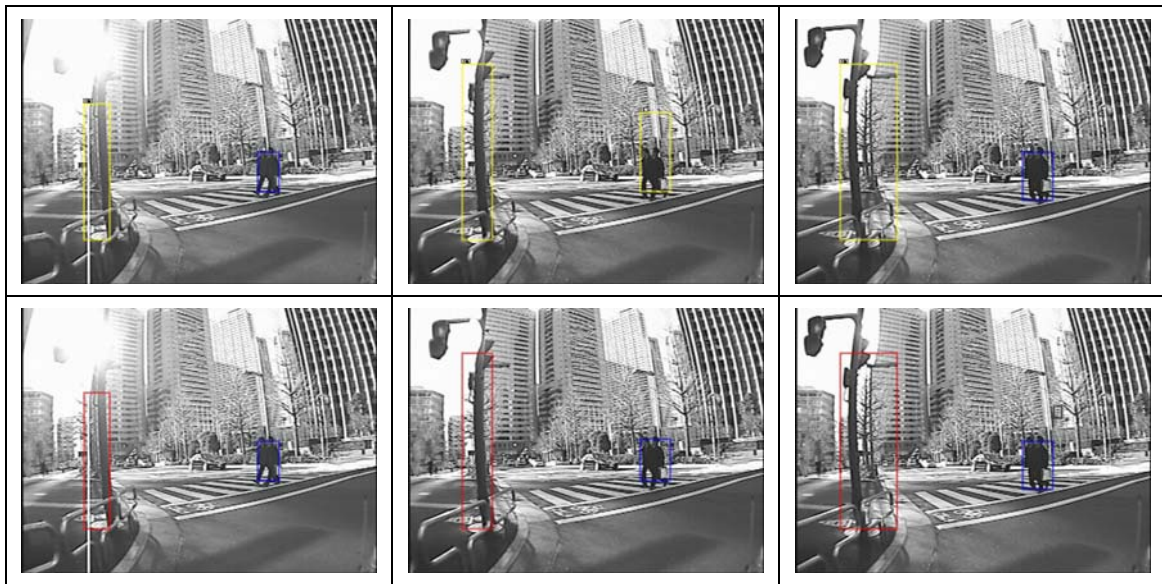


図 6.4.1 ROI 分割アルゴリズム適用例 ケース 1 (上段：適用前 下段：適用後)

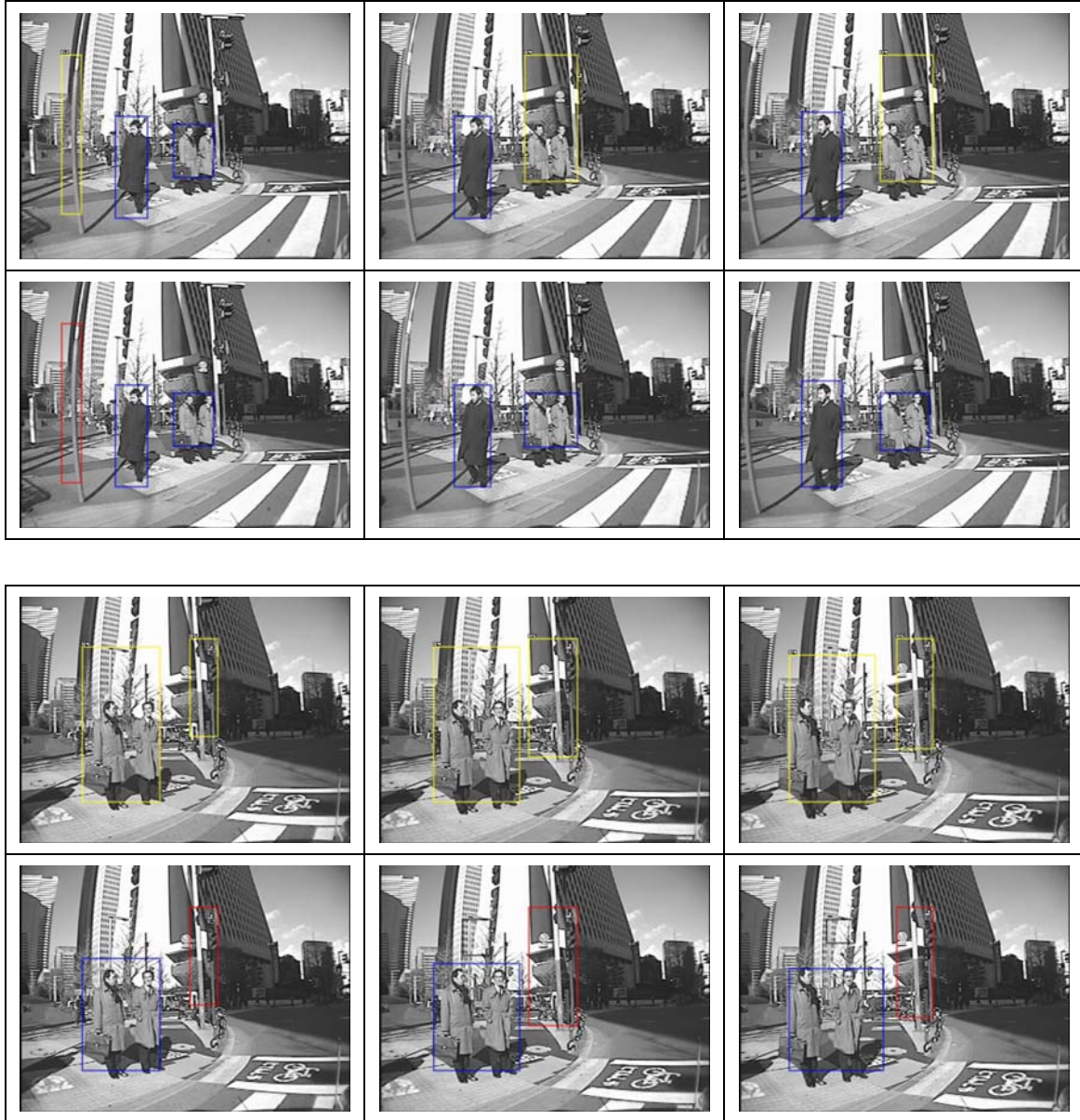


図 6.4.2 ROI 分割アルゴリズム適用例 ケース 2 (上段 : 適用前 下段 : 適用後)

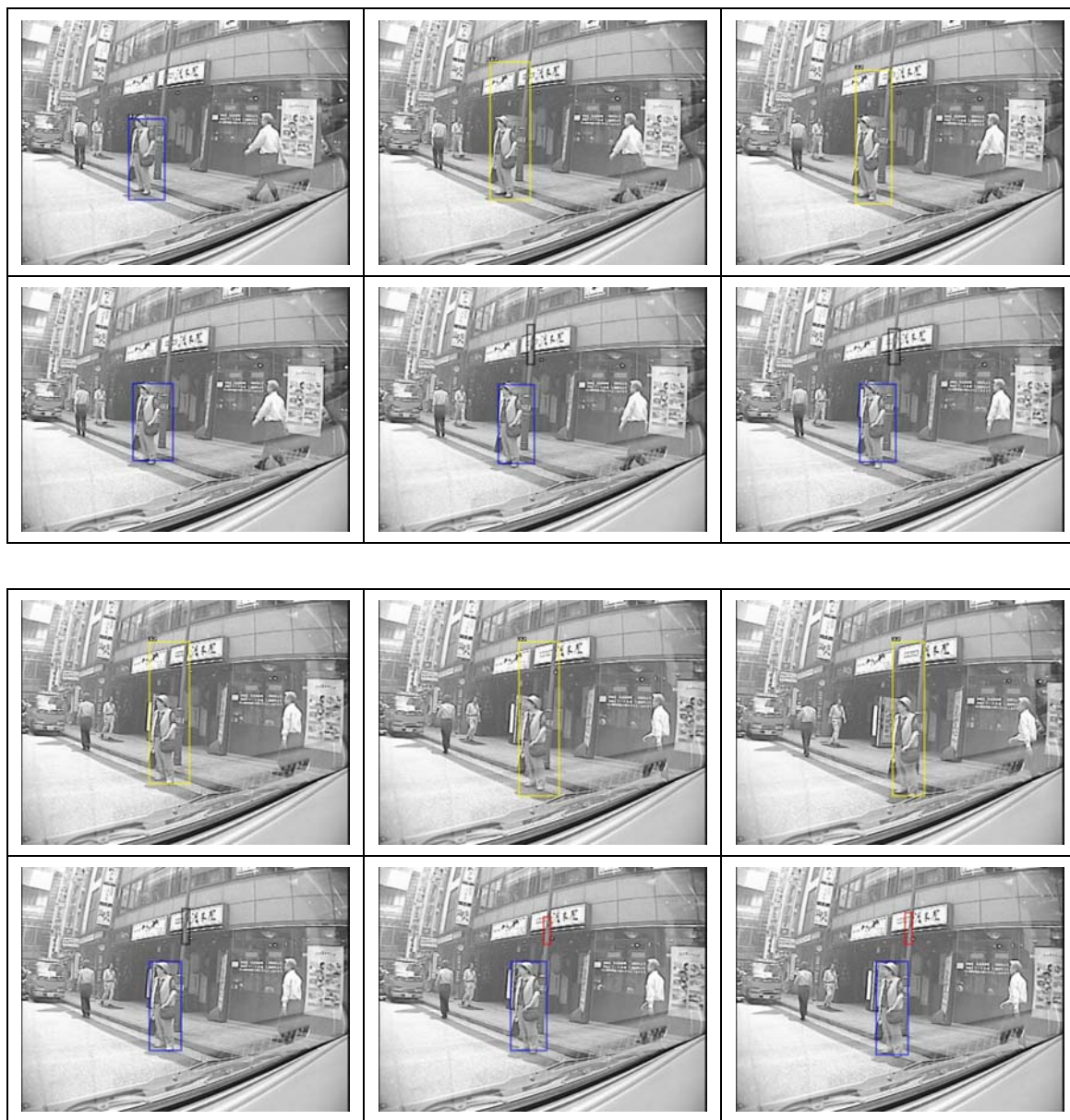


図 6.4.3 ROI 分割アルゴリズム適用例 ケース 3 (上段 : 適用前 下段 : 適用後)

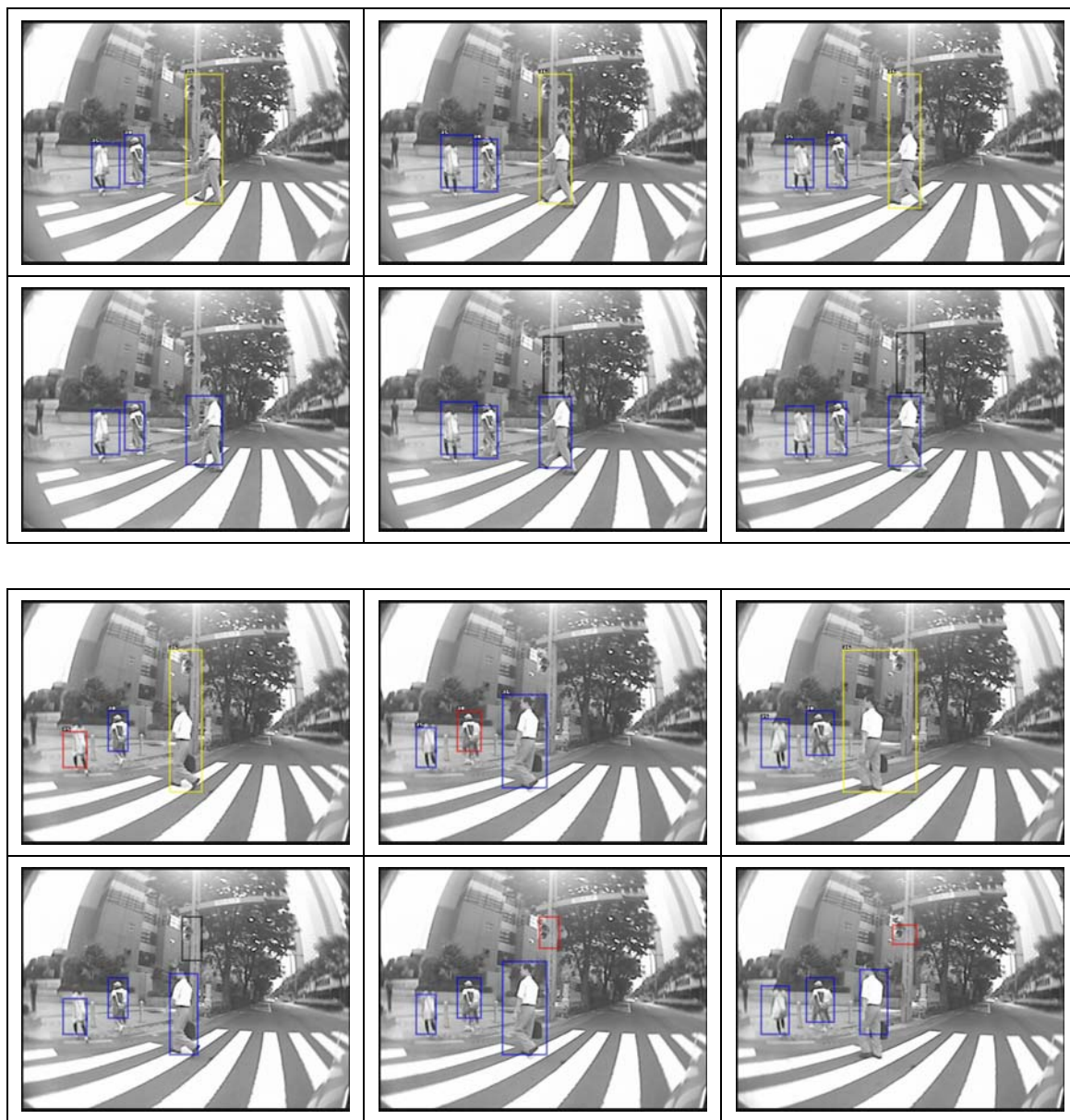


図 6.4.4 ROI 分割アルゴリズム適用例 ケース 4 (上段 : 適用前 下段 : 適用後)

第 7 章 実験結果と今後の課題

前章までは本研究の基本的なアルゴリズムと，判別性能を上昇させる様々な工夫，また相対軌跡の導出や ROI 情報の修正などについて述べた．本章では本研究のアルゴリズムを様々なシーンに対し適用し，その判別性能や相対軌跡などについて論ずる．また結果から考えられる考察，課題などを述べる．

7.1 実験結果

7.1.1 対象シーンについて

第1章や第2章に述べたように、本研究は直進時や交差点での右左折時などの歩行者検出を目的にしている。よって性能評価を行うシーンもそのような状況を想定して収集した。なお、収集は新宿・渋谷・お台場・豊洲などの非と・車ともにある程度の通行量がある交差点、直進道路で行った。街中を走行し撮影しつつ図 1.2, 図 1.3 に示されるカメラ設置、状況で撮影映像を保存し、交差点における周辺監視に関しては交差点に進入する 10 秒程度前から、直進道路における前方監視についても横断歩道前 10 秒ほどまえからシーンを切り出し、本研究のアルゴリズム処理を行った。

本研究では図 1.3 にあるように、車体の中心に設置し前方を向くカメラは高速度で直進道路を走行中、前方を横切る歩行者を検出し衝突防止をする目的に用いている。そのため前方歩行者監視の際には遠方が撮影できるように画角が 20° と 30° のレンズを用いている。また一方で車両側部に取り付けられているカメラは、車両周辺監視を行い巻き込み事故防止をする目的に用いているので、単一のカメラで車両周辺をカバーできるように前方監視カメラより広い画角 60° , 75° , 100° のレンズを用いて撮影を行った。

シーン総数は 60 シーンで、10fps の画像を用いており総フレーム数は時間にすると約 1 時間ほどの分量に対し性能評価を行った。登場歩行者は 60 シーンでおおよそ 400 人、約 9000 フレームの歩行者が登場している。なお、第4章で述べた学習データは評価に用いたシーンとは全く異なるシーンから抽出したデータで、両者が重複する事はない。次項で評価手法と結果を示す。

7.1.2 評価手法および性能評価

表 7.1 は本研究の結果である。表中の“正解フレーム数”とはシーン中に登場する歩行者のフレーム数を数えたものを全てのシーンについて総計した総フレーム数を意味し、いわゆる真値である。ここでいう“フレーム”とはそれぞれの歩行者が登場するフレームについて各々足し合わせたものである。つまりある歩行者Aが 201 フレーム目から 210 フレーム目に登場しており、歩行者Bが 206 フレーム目から 210 フレーム目に登場している場合には歩行者Aについて $210 - 201 + 1$ で 10 フレーム、歩行者Bについて $210 - 206 + 1$ で 5 フレームとなり、このシーンの 201~210 フレームにおける歩行者の“正解フレーム数”は $10 + 5$ で 15 フレームとなる。表中の“正解フレーム数”はこの計算を評価シーンすべてについて総計したものである。

表 7.1 中“正報数”とは評価シーン中に登場する歩行者について、本研究のアルゴリズムが正しく“歩行者”と判定できたフレーム数を表す。そのため“欠報率”とは対象 ROI 内に歩行者が存在しているのに、アルゴリズムが歩行者ではなく“非歩行者”と判定してしまった率を表す。“欠報率”が高いシステムは見落としが多いシステムとなるので、そもそも意味のないシステムとなってしまう。

“発報数”とはアルゴリズムが“歩行者”と判断したフレーム数のことであり、これには実際は歩行者であり“歩行者”と判断されたケースや、逆に実際は非歩行者でありながら“歩行者”と判断されたケース、いわゆる“誤報”も同様に含まれている。“誤報数”とはこの歩行者が存在しない ROI でありながらアルゴリズムが歩行者と判定した“誤報”の全評価シーンに対する総計である。そして“誤報率”とはアルゴリズムの“発報数”に対する、“誤報”の割合である。この“誤報率”はいわゆる登場するすべての非歩行者 ROI に対する誤報数の割合である誤検出率とは異なるので留意しておきたい。実システムの運用を考慮するときは、あくまでシステムがアラームを出した数のうち、本当に正しい発報と間違っている発報の割合である“誤報率”が重要なのであり、“誤検出率”はそれほど重要ではないといえる。なお、欠報率と誤報率の算出法を下に示す。

$$\text{欠報率(\%)} = \frac{\text{正解フレーム数} - \text{正報数}}{\text{正解フレーム数}}$$












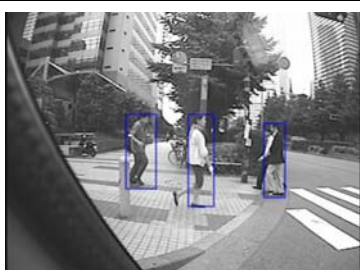
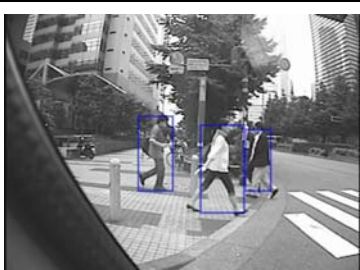


$$\text{誤報率(\%)} = \frac{\text{誤報数}}{\text{発報数}}$$

第 6 章における ROI 修正アルゴリズムなし処理をかけた結果と、修正アルゴリズムを実装し、処理をかけた場合の結果について表 7.1 に示す。

図 7.1.1～7.1.4 は評価シーンに対し本研究のアルゴリズム処理をかけた結果画像である。画像中青枠が歩行者と判定された ROI を、赤枠が非歩行者と判定された ROI を示す。また黒枠は判定準備段階であることを示す。

表 7.1 アルゴリズム性能評価

	正解フレーム数	発報数	
		正報数	誤報数
セマンティックHOGなし ROI 修正アルゴリズムなし	9270	9113	
		8493 (欠報率 = 8.38%)	835 (誤報率 = 8.96%)
セマンティックHOGあり ROI 修正アルゴリズムあり	9270	9292	
		8911 (欠報率 = 3.87%)	381 (誤報率 = 4.10%)

		
64 フレーム	66 フレーム	68 フレーム
		
72 フレーム	79 フレーム	83 フレーム
		
85 フレーム	92 フレーム	95 フレーム
		
95 フレーム	101 フレーム	106 フレーム
		
110 フレーム	114 フレーム	117 フレーム

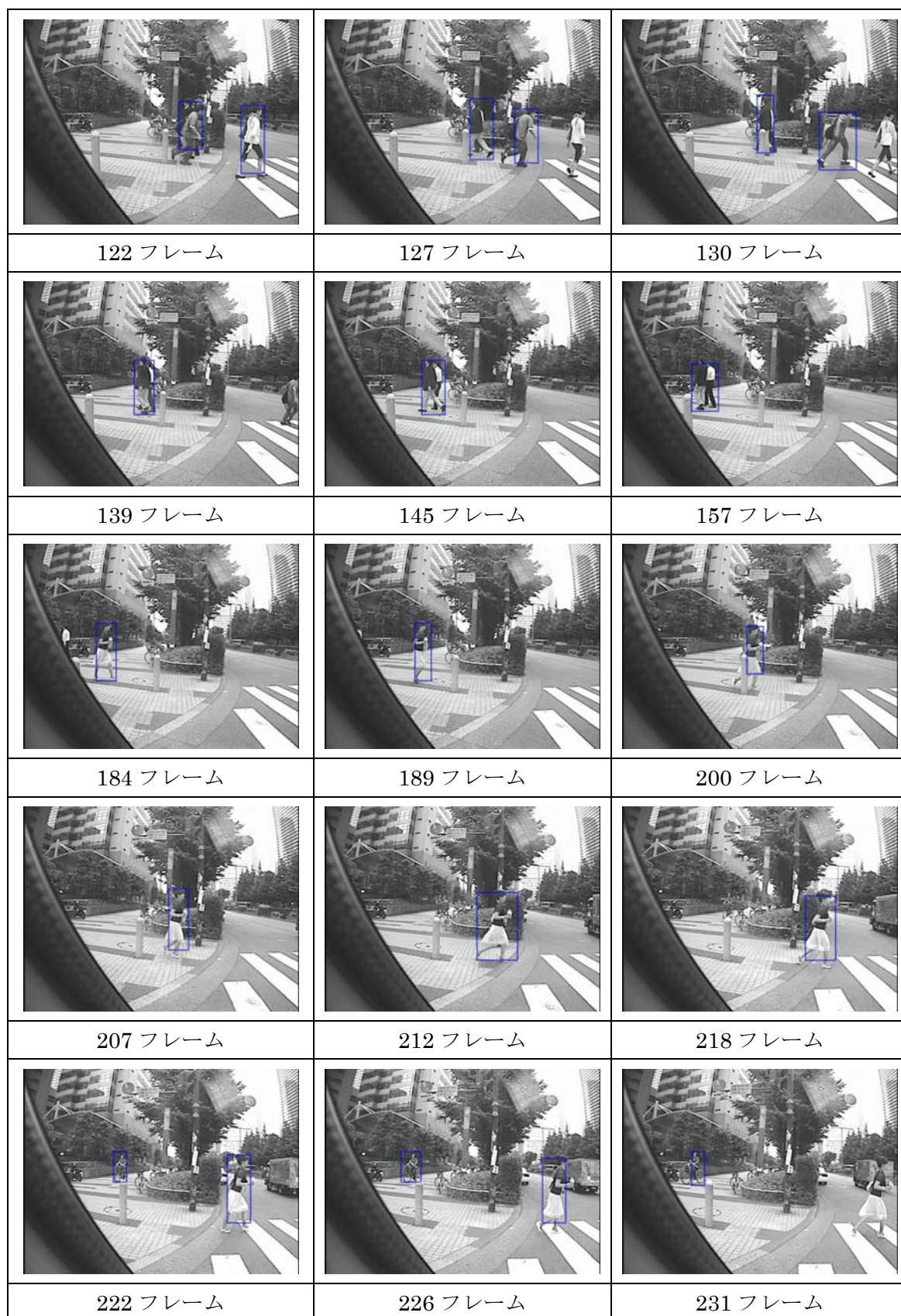














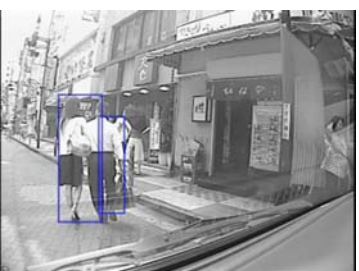


図 7.1.1 実験結果画像 シーン1：左折 左カメラ 75° レンズ使用

		
590 フレーム	610 フレーム	625 フレーム
		
631 フレーム	658 フレーム	665 フレーム
		
680 フレーム	720 フレーム	735 フレーム
		
746 フレーム	759 フレーム	772 フレーム
		
786 フレーム	811 フレーム	834 フレーム

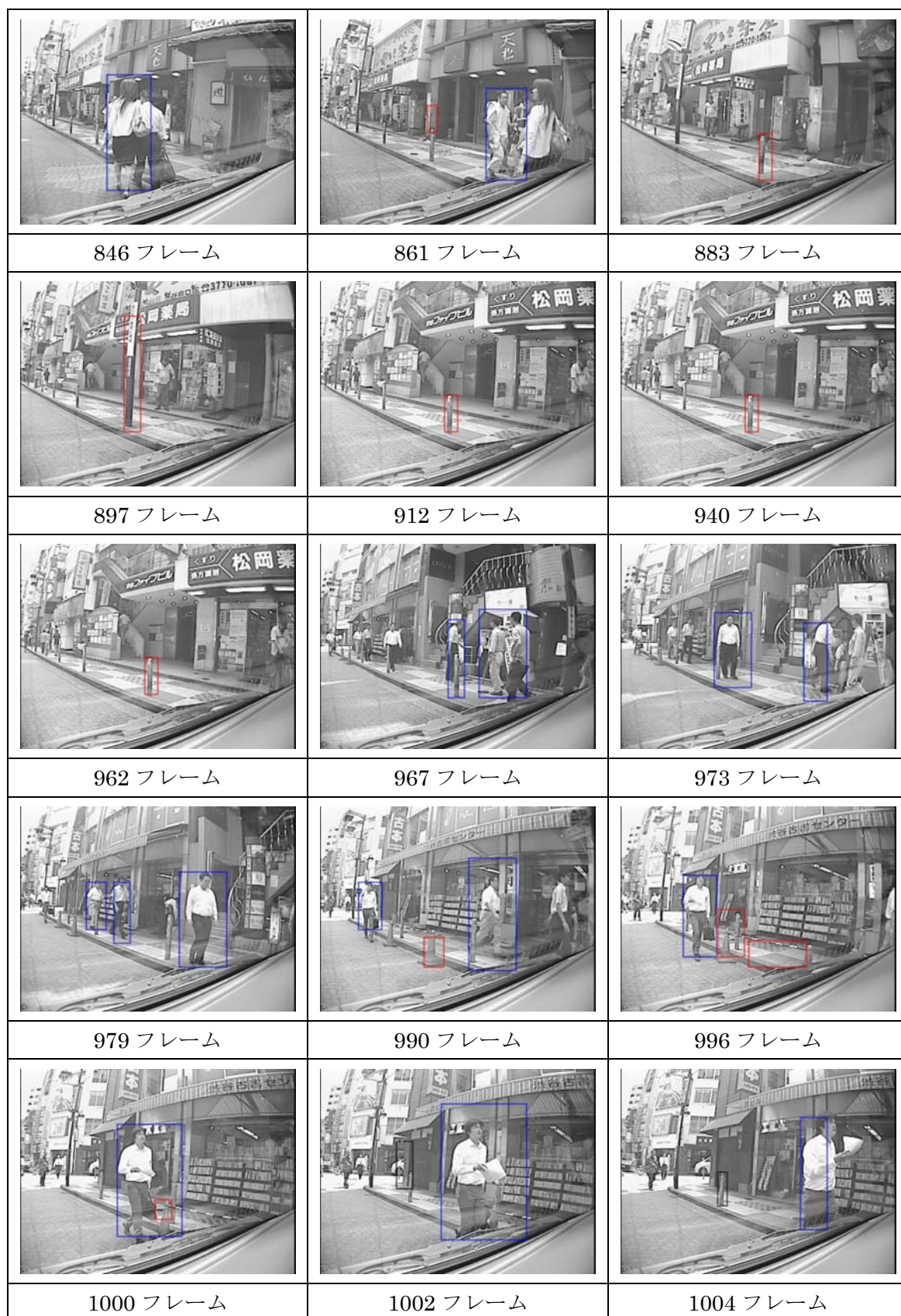







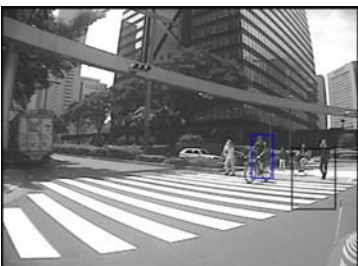
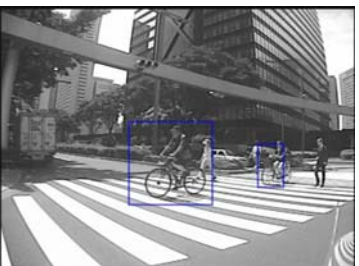
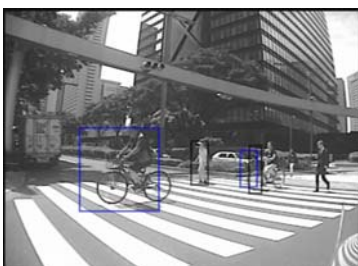
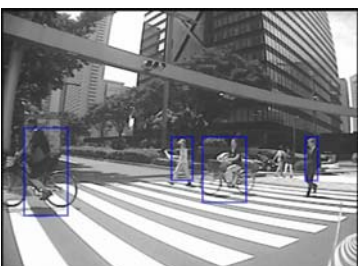

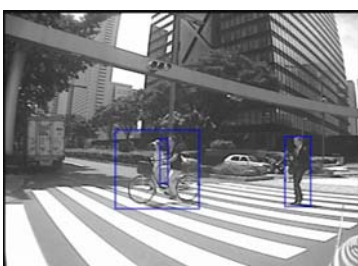
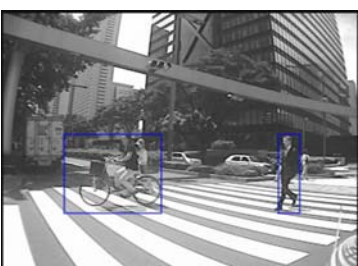
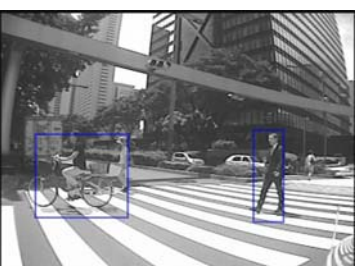


図 7.1.2 実験結果画像 シーン2：直進 右カメラ 60° レンズ使用

		
126 フレーム	136 フレーム	139 フレーム
		
165 フレーム	171 フレーム	175 フレーム
		
180 フレーム	186 フレーム	196 フレーム
		
199 フレーム	204 フレーム	207 フレーム
		
210 フレーム	213 フレーム	216 フレーム

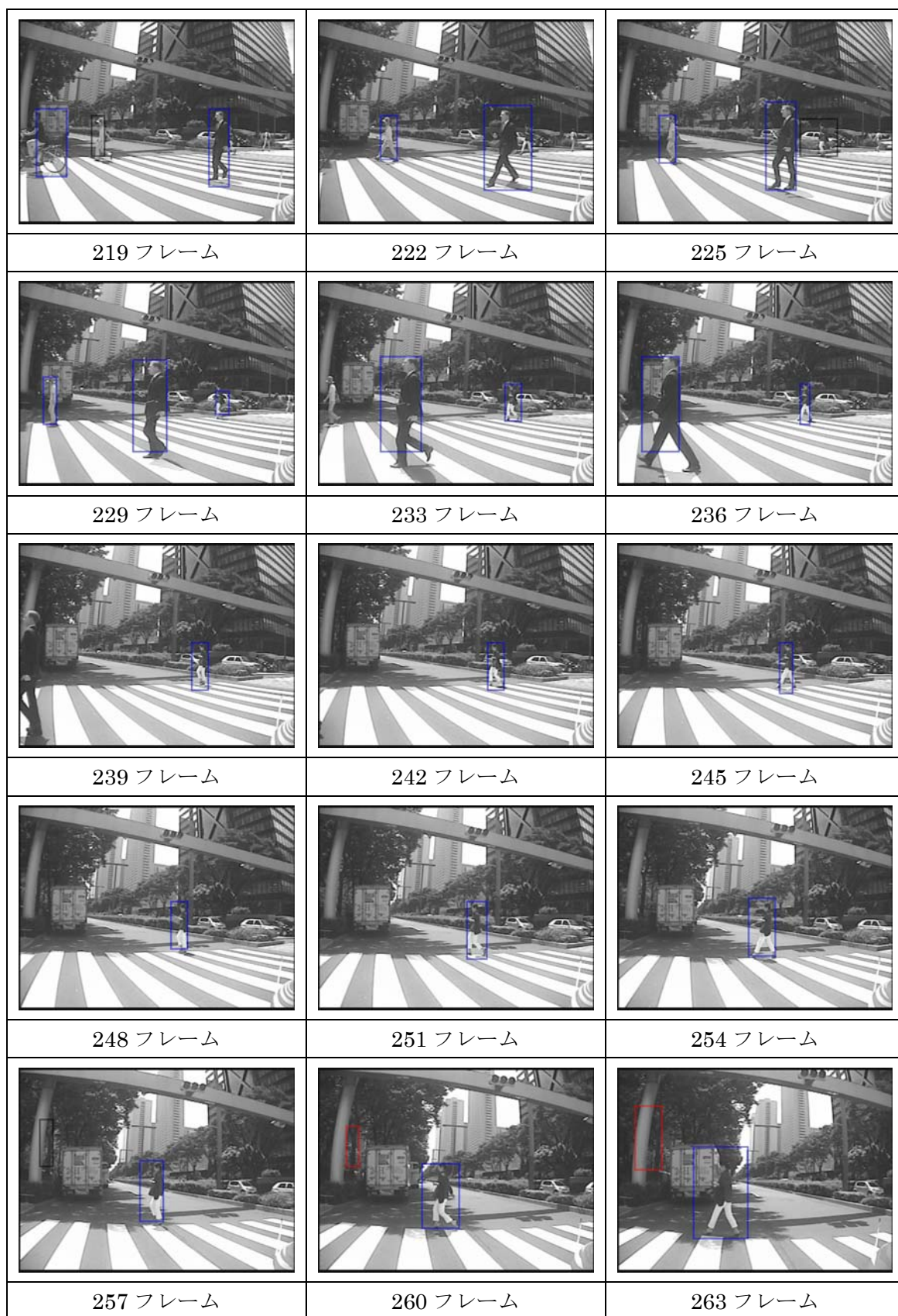
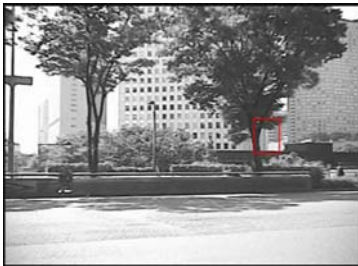
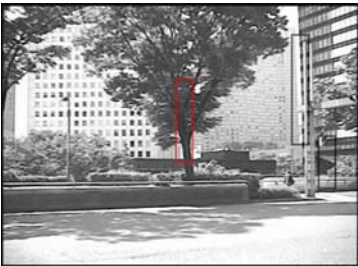
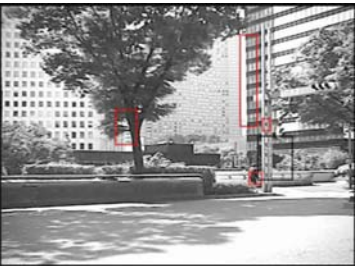
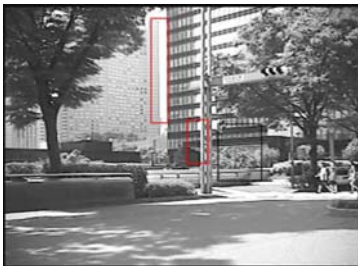







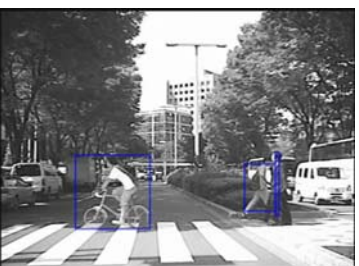

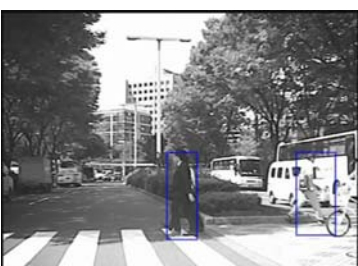
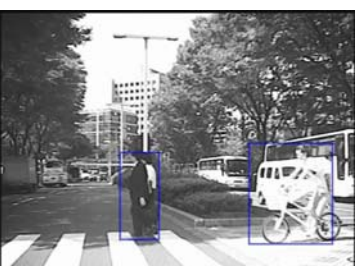


図 7.1.3 実験結果画像 シーン3：左折 左カメラ 75° レンズ使用

		
140 フレーム	149 フレーム	155 フレーム
		
165 フレーム	176 フレーム	185 フレーム
		
190 フレーム	197 フレーム	201 フレーム
		
205 フレーム	211 フレーム	216 フレーム
		
222 フレーム	227 フレーム	231 フレーム

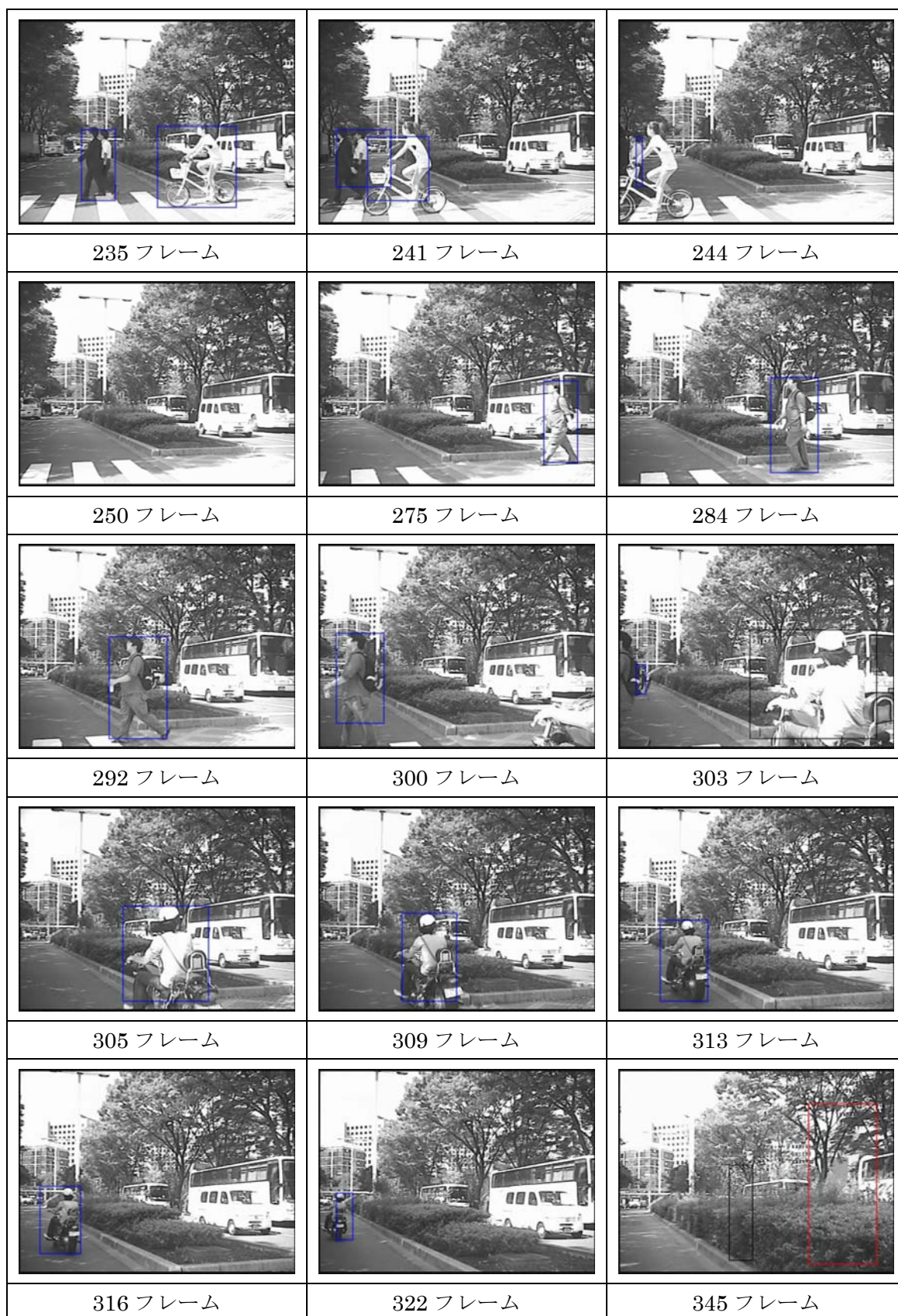


図 7.1.4 実験結果画像 シーン4：右折 右カメラ 30° レンズ使用

7.2 結果考察

表 7.1 を見ると、第 4 章で述べたセマンティック HOG 特徴量による判別や、第 6 章で述べた ROI 情報修正アルゴリズムを導入した結果、欠報率が 8.38% から 3.87% へ、誤報率が 8.96% から 4.10% へそれぞれ減少した。これはセマンティック HOG 特徴量判別により導入前まで誤って歩行者と検出され誤報になっていたオブジェクトが、正しく非歩行者と判定された事と、ROI 情報修正アルゴリズムの導入により、今まで欠報となっていた歩行者が正しく歩行者と判定されるようになった事、これら 2 つのアルゴリズムの追加・改良により結果向上につながったと思われる。

また ROI 情報修正アルゴリズムにより欠報率の低下のみならず、歩行者軌跡がより正確に取得できるようになる効果も得られた。第 5 章で述べたように、本研究は ROI 情報の下辺を用いて歩行者の車両に対する相対軌跡を取得することができる。本研究は将来的に、この相対軌跡を用いて車両に対する衝突危険性を算出し、それにより発報すべきかを決定するシステムへ発展する事を想定している。そのため歩行者検出の精度に関して問題なかったとしても、相対軌跡が正確に算出できないことにはシステムとして実用レベルに引き上げることが出来ない。以上の事から相対軌跡が正確に取得できるようになったことは将来的に非常に有益なことであると言える。

評価シーンには、図 7.1.2 の 772 フレーム・786 フレームの画像中央付近に登場する女性 2 人組のような、そもそも時空間 MRF モデルが検出していないような歩行者がその他の評価シーンにも登場する。本研究ではこれらのような人物も欠報として性能評価結果を算出している。時空間 MRF モデルは背景の動き量を線形式に近似し、その近似値から外れた動き量をもつ領域をオブジェクトとして検出するのは第 3 章で述べたとおりである。しかしこれらのように背景付近に立って静止している人物は、背景との動き差分がないために非常に検出しづらい、よってここでは未検出となっている。しかし安全運転支援の視点立って考えると、背景付近に立って動かない人物というのはそもそも支援の必要がないとも言え、これらの人物を欠報に入れないと考えると、欠報が 3.87% から 2.82% に低下され結果が上昇する。

7.3 まとめと今後の課題

まず本研究で用いた歩行者検出アルゴリズムについて説明する．まず動き検出アルゴリズムによって検出された領域を ROI として抽出し，抽出された ROI を時空間 MRF モデルによってトラッキングする．ここで種々のエラーにより領域決定が正確に行われていないと判断されたときには，現フレームまでの ROI 情報を元に ROI 情報の修正が行われる．次にそれらの ROI の位置と高さから ROI の現実での高さを算出し，その値が歩行者として不適な ROI はその時点で非歩行者オブジェクトと判定する．歩行者として適した高さを持つ ROI は，HOG/Fisher 判別器にかけられる．最後にセマンティック HOG 特徴量を用いて特徴的なテクスチャ情報を持つ非歩行者オブジェクトを除外する．

本研究では，用いるカメラを限定しない単眼での歩行者検出技術の開発・評価を行った．具体的には 20° から 100° のレンズを用いて様々な実際のシーンを用いて評価を行い，高い成果を達成した．さらに本アルゴリズムは歩行者検出を行うシーンを変更しても，わずかなカメラキャリブレーションを行うことで即座に対応する事ができる．カメラの解像度と視野角はトレードオフの関係，つまり検出対象物との距離と検出可能視野の角度はトレードオフの関係にある．例えば直線走行時の安全運転支援では，時速 50km 前後の比較的高速走行時における前方飛び出しなどの事故防止を目的としているため，狭い視野で遠方の歩行者を検出することが重要である．一方交差点での右左折時では，直進時に比べ低速で走行するが，車両周辺を広視野で歩行者の検出を行わなければならない．このように対象とするシーンにより用いるカメラ・レンズは変更することが予想されるが，本研究のアルゴリズムはそのような変更に対し，わずかな設定の変更で前方監視も交差点での巻き込み監視も，同一のアルゴリズムを用いて行うことができる．

ROI 情報修正アルゴリズムにより高い精度で歩行者の相対軌跡を取得する事が可能になった．実際に歩行者検出技術を用いて安全運転支援システムを実装した際に，歩行者の危険度を測定するためにも相対軌跡の取得は非常に重要だと思われる．自車両のロール運動などが相対軌跡の取得に影響を及ぼすかを調査するために，第 5 章において本アルゴリズムによる相対軌跡の正確性について実験したが，通常のアスファルト面を走行している限り大きな影響はないと判断できた．

本研究は車載単眼カメラによる道路上の歩行者検出を目的にしており，検出精度に関しては同分野のその他研究と比較しても高精度な性能を達成した．今後の課題・発展としては，第一に相対軌跡を用いた歩行者の衝突危険性の算出が挙げられる．本研究では評価シーン中に登場するすべての歩行者に対し検出を行っているが，実システムを考えると検出されたすべての歩行者に対し発報を行うのは得策とは言えない．特に日本の交差点のように大量の歩行者が存在するような場所では，各歩行者に対しいちいち発報するわけにはいかない．そのため各歩行者の衝突危険性を算出し，危険性の高い歩行者のみ発報を行うよ

うなシステムが考えられる。その衝突危険性の算出に、相対軌跡を用いることが今後の課題といえる。またその他にも、検出結果や相対軌跡のさらなる精度向上も挙げられる。具体的には Controller Area Network データ(CAN データ)を用いることが予想される。CAN データとは車両の速度・エンジン回転数・ブレーキなど車両運転にまつわる様々な情報をデータ化したものである。第 5 章での相対軌跡の正確性の検証では大きな問題はなかったが、路面状況の具合によっては問題になる事は考えられる。CAN データを引き出し車両姿勢の情報などをもとに補正を行い更に精度を向上させることは充分可能であるといえる。

謝辞

本研究を進めるにあたり，御指導，御助言をいただきました上條俊介准教授に深く感謝いたします。研究の基礎や進め方，研究に対する姿勢など大学院生活の基本的な事柄から，研究者とはどうあるべきかなど今後の人生においても非常に重要なことなど，様々なことをご指導頂きました。当初右も左もわからなかった僕を辛抱強くご指導していただき，研究者の端くれとして自覚を持たせて頂きました。本当にありがとうございました。

STARC 研究員の方々には目標を定めていただき，研究に張り合いができました。厳しい性能要求には困難とを感じる時もありましたが，結果的に壁を乗り越える方法や，乗り越えたときの喜びを知ることが出来ました。

研究員の藤村さんには ITS の分野に疎かった自分に，プログラミングやツールの使い方，研究の目的など様々なことを教えていただきました。

小野寺先生には英語論文の指導や発表練習に付き合ってくださいなど，本来の業務を越えていろんな指導をしていただきました。また，小野寺先生のおかげで研究室内の雰囲気も暗くならず，明るく研究に打ち込むことが出来ました。ありがとうございます。

秘書の三輪さんには伝票整理や書類処理などの面で大変お世話になりました。また就職活動の時期にはスーツを補修していただきまして，本当にありがとうございました。

後輩の金君と日坂君には，いろんなデータ整理や論文の校正など面倒くさい仕事を頼みすぎてしまったかもしれません。迷惑をかけたとは思いますが，頼んだ地味な仕事をこなした事は来年きっと役に立つと思うので頑張ってください。

研究室の皆様，友人や家族など様々な人のおかげで研究に集中し，まとめることが出来たのだと思います。最後に重ねて御礼申し上げます，本当にありがとうございました。

平成 23 年 2 月 9 日

発表文献

- [1] Yuuki Shibayama, Shunsuke Kamijo, Kaichi Fujimura, "Pedestrian Tracking and Detection with On-board Multi View Angle Cameras," 17th ITS World Congress 2010.
- [2] 柴山悠毅, 藤村嘉一, 上條俊介, “車載カメラを用いた車両周辺の歩行者軌跡の取得,” 第 9 回 ITS シンポジウム 2010, 1-D-02, 2010.
- [3] 柴山悠毅, 藤村嘉一, 上條俊介, "画像認識による車両周辺監視技術の開発," 第 16 回画像センシングシンポジウム, No.IS4-12, 2010

参考文献

- [1] 内閣府，“平成22年度交通安全白書”，
<http://www8.cao.go.jp/koutu/taisaku/index-t.html>
- [2] 経済産業省，“IT新改革戦略の推進”
http://www.meti.go.jp/policy/it_policy/it-strategy/index.html
- [3] 国土交通省道路局 ITS ホームページ, <http://www.mlit.go.jp/road/ITS/j.html/>
- [4] 安全走行支援サービス参宮橋社会実験ホームページ
<http://sangubashi.ahsra.or.jp/pc/index.html>
- [5] N.Sumiya, K.Fujihira, S.Kamijo, “Incident Detection System by Sensor Fusion Network Employing Image Sensors and Supersonic Wave Sensors”, IEEE Conference on Intelligent Transportation Systems, Paper No.189, 2006
- [6] UTMS 社会法人新交通管理システム協会，“安全運転支援システム”
<http://www.utms.or.jp/japanese/system/dsss.html>
- [7] C.Wang, C. Thrope, “LADAR-based detection and tracking of moving objects from a ground vehicle at high speeds”, Proceedings of the IEEE IV Symposium, June, 2003
- [8] S. Gidel, P. Checchin. “Pedestrian detection method using a multilayer laserscanner: Application in urban environment.” IROS.IEEE/RSJ International onference on, pages 173–178, Sept. 2008
- [9] C. Papageorgiou, T. Poggio, “A trainable system for object detection,” International Journal of Computer Vision 38 (1) (2000) 15–33.
- [10] D. M. Gavrila, and S. Munder, “Multi-cue pedestrian detection and tracking from a moving vehicle,” Int J. Comput. Vis., 2007.
- [11] N. Dalal, B. Triggs, “Histograms of oriented gradients for human detection,” IEEE Conference on Computer Vision and Pattern Recognition 2 (2005) 886–893.
- [12] M. oren, C. Papageorgiou, P. Sinha, E. Osuna, and T. Poggio., “Pedestrian detection using Wavelet template,” In Proceedings of IEEE CVPR 97, pages 193-199,1997.
- [13] S. Munder, C. Schnorr, D. M. Gavrila, “Pedestrian Detection and Tracking Using a Mixture of View-Based Shape–Texture Models,” IEEE Transactions on Intelligent Transportation Systems, Volume: 9, Issue: 2, page(s): 333-343 , June 2008.
- [14] R. Cutler, L. Davis, “Robust real-time periodic motion detection: analysis and applications,” IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (8) (2000) 781–796.
- [15] H. Sidenbladh, “Detecting human motion with support vector machines,” Proceedings of the 17th International Conference on Pattern Recognition, 2:188-191, 2004.

- [16] C. Curio, J. Edelbrunner, and T. Kalinke, Christos Tzomakas, and Werner von Seelen, “ Walking Pedestrian Recognition.,”IEEE Transaction on Intelligent transportation Systems, Vol.1, No. 3, September 2000.
- [17] P. Viola, M.J. Jones, and D. Snow, “Detecting pedestrian using patterns of motion and appearance”, IEEE International Conference of Computer Vision, 2:734-741,2003.
- [18] Elzein, H. Lakshmanan, S. Watta, P. , “A motion and shape Based pedestrian Detection algorithm,” Proceedings. IEEE Intelligent Vehicles Symposium, 2003.
- [19] Broggi A, Cerri P, Ghidoni S, Grisleri P, Jung HG. “A new approach to urban pedestrian detection for automatic braking”, IEEE Trans Intell Transp Syst. 2009; 10(4):594-60.
- [20] C. Premebida, O. Ludwig, “Lidar and vision based pedestrian detection system”.Journal of Field Robotics, 26(9), 2009
- [21] N. Kaempchen, K. Dietmayer,” Fusion of Laserscanner and Video for Advanced Driver Assistance Systems” 11th World Congress on ITS Nagoya, 2004
- [21] S. Kamijo, M. Sakauchi, “Simultaneous Tracking of Pedestrians and Vehicles in Cluttered Images at Intersections,” 10th World Congress on ITS, Madrid, November.2003, CD-ROM
- [22] B. Sen, K. Fujimura, S. Kamijo, “Pedestrian Detection by On-board Camera Using Collaboration of Inter-layer Algorithm”, ITSC2009 pp588-595, October 2009, St.Louis.
- [23] S.Kamijo, K.Fujimura, Y.Shibayama, “Pedestrian Detection Algorithm for On-board Cameras of Multi View Angles” Proceedings. IEEE IVSymp, 2010.
- [24] LEAR Datasets & Images <http://lear.inrialpes.fr/data>