

ドライバアシストシステムのための画像分割技術による 移動物体検知

Image Segmentation-based Moving Object Detection for Driver Assistance System

西 垣 守 道*

Morimichi NISHIGAKI

Yiannis ALOIMONOS**

要 旨

ABSTRACT

自動車の進行方向上にある物体を検知することは、衝突防止に有効である。特に、自動車や歩行者のような移動物体とは衝突の可能性が高く、検知の必要性は高い。ビジョンセンサには、実際の運転状況で遭遇するさまざまな物体の検知が要求される。しかしカメラと物体は、どちらも独立に動いているため、車両に取り付けたカメラで、自律的に動いている物体を検知するのは容易ではない。移動物体の輪郭を鮮明に検知することは、特に難しかった。そこで、我々は移動物体の輪郭を鮮明に検知できる、二つのアルゴリズムを開発した。第1のアルゴリズムは、画像上の動きとエピポーラ制約との差異から、カメラと異なる方向に移動している物体を検知する。第2のアルゴリズムは、ステレオ視差と運動視差との差異から、カメラと同じ方向に移動している物体を検知する。これらのアルゴリズムでは、画像分割技術(イメージセグメンテーション)を用いることで、物体の輪郭を鮮明に検知できた。この手法で移動物体の輪郭を鮮明に検知できることを、実車を用いた実験により実証した。

The detection of objects in front of ego-vehicle is an effective means to help prevent collisions. Since the potential of collision with moving objects, such as other vehicles and pedestrians, is relatively high, it is important to detect these moving objects in particular. Vision sensors are employed to detect the various objects encountered in actual driving situations, but because the camera and the objects are moving independently, it is not easy to detect independently moving objects using vehicle-mounted cameras. The detection of sharp boundaries for moving objects has been especially challenging. We have therefore developed two algorithms to detect sharp boundaries for moving objects. The first of these algorithms detects objects moving in a different direction to the camera by identifying inconsistency between motion in the images and the epipolar constraint. The second algorithm detects objects moving in the same direction as the camera by identifying inconsistency between stereo disparity and motion disparity. The use of image segmentation technology in these algorithms has made it possible to detect sharp boundaries for moving objects. The ability to detect sharp boundaries using these methods was verified in tests using actual vehicles.

1. 緒言

Intelligent Transportation Systems(ITS)の中で、Advanced Driver Assistance System(ADAS)は、活発に研究がおこなわれている分野である。車両周辺にある自動車、歩行者、レーンマーカ、信号のような物体を検知す

るセンサは、ADASの重要な構成部品である。ビジョンセンサは、これらさまざまな物体を検知できるセンサとして期待されている。安全運転支援の観点からは特に、衝突する可能性の高い、自動車や歩行者などの自律移動物体を検知する必要がある。しかし、自律移動物体を画像から検知するのは、容易ではない。なぜなら、移動する車両に

* HONDA R&D EUROPE(DEUTSCHLAND)GMBH

** University of Maryland

取り付けカメラで撮影した画像では、静止物体も、移動物体と同様、動いて見えるためである。そのような動画から、いかにして移動物体領域を検知するかが課題である^{(1), (2)}。移動物体が、カメラの移動方向と異なる方向に移動している場合には、静止物体の見かけの動きに関する制約条件を表す、エピポーラ制約⁽³⁾を利用すれば、1台のカメラ（単眼カメラ）で、移動物体を検知できる。移動物体がカメラの移動方向と同じ方向に移動している場合には、1台のカメラでは検知不可能で、2台のカメラ（ステレオカメラ）が必要とされ、アルゴリズムもより複雑になる。そこで、これらの手法に適用する、二つのアルゴリズムを開発した。自動車応用では物体の幅や大きさの検知も求められる。この手法では、画像を有意義な部分領域に分割する、画像分割技術（イメージセグメンテーション）を用いて、物体の輪郭を検知する。

2. アルゴリズム

Ogale らは画像による移動物体検知を、カメラの動きなどとの関係から、三つのクラスに分類した⁽⁴⁾。カメラの動く方向と別の方向に移動する物体は、クラス1に分類される。この場合、物体の動きが、エピポーラ制約を満たしているかどうかによって、移動物体か否かの判定ができる。エピポーラ制約は、移動物体がカメラと同一方向に動いている場合は使えない。カメラの動きと同一方向に移動していて、さらに一部が静止物体（遮蔽物）によって隠れているような移動物体はクラス2に分類される。この場合は、遮蔽による前後関係と運動視差に矛盾がないかを調べることで、自律移動物体を検知できる。カメラと同一方向に移動するが、遮蔽物がない移動物体は、クラス3に分類される。この場合、運動視差とステレオ視差によって得られる前後関係に矛盾がないかを調べることで検知できる。

前述したクラス2の検知手法では、移動物体とそれを部分的に遮蔽する静止物体との距離によっては検知できない場合もある。しかし、クラス3の移動物体を検知する方法を使えば、どのクラスに属する移動物体でも検知できる。

そこで、この分類と検知原理に基づき、車載カメラで移動物体を検知する、二つのアルゴリズムを開発した。第1のアルゴリズムは、クラス1に属する移動物体を単眼カメラで検知する。第2のアルゴリズムは、クラス2とクラス3を含むより一般的な移動物体をステレオカメラで検知する。第1のアルゴリズムは、第2のアルゴリズムより単純であるが、検知できる移動物体は限られる。第2のアルゴリズムは、より一般的な移動物体を検知できるが、ステレオカメラが必要となる。二つのアルゴリズムは検知したい対象に応じて使い分ける必要がある。これら二つのアルゴリズムと交通状況とを対応させるため、第1のアルゴリズムを交差移動物体検知アルゴリズム、第2のアルゴリズムを並走移動物体検知アルゴリズムと呼ぶ。つまり、交差移動物体検知しか求められないシステムに適用する際には、単眼カメラで実現できる、よりシンプルな、第1のアルゴリズムを活用することができる。並走移動物体検知も求められるシステムに適用する際には、ステレオカメラが必要で、よりアルゴリズムとシステム構成が複雑化するが、第2のアルゴリズムを活用できる。

この章では、まず2.1.節、2.2.節で、アルゴリズムの要点を解説し、2.3.節、2.4.節でそれぞれ、交差移動物体検知アルゴリズムと、並走移動物体検知アルゴリズムの詳細を説明する。

2.1. 移動物体検知原理

交差移動物体検知アルゴリズムと、並走移動物体検知アルゴリズムでは、検知原理が異なる。

交差移動物体検知アルゴリズムにおける、検知原理を表す模式図を、Fig. 1に示す。この図では画像上に自車が直進している場合のエピポーラライン⁽³⁾が青線で描かれている。この場合、エピポーララインは静止物体の移動方向を表している。エピポーララインが交差する点が、Focus of Expansion (FoE)であり、すべての静止物体は、画像上をFoEからエピポーララインに沿って各方位に移動するように撮影される。エピポーラ制約とは、このように静止物体の画像上での見かけの動きに、幾何学的制約があるこ

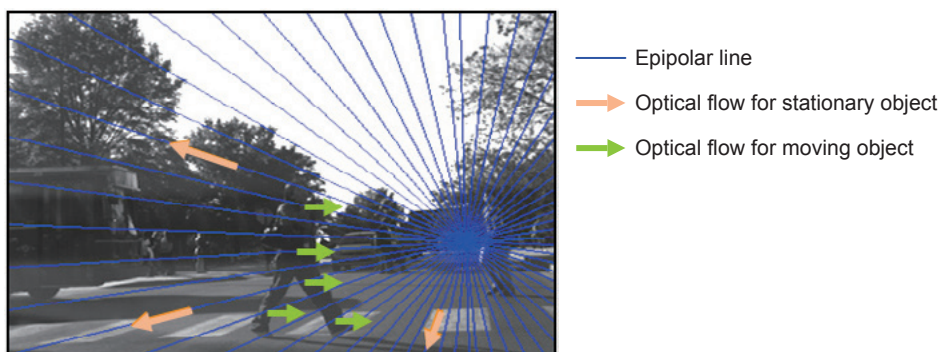


Fig. 1 Principle of detection of moving objects for intersecting objects

とを表すもので、ある画像上の点の動きが、エピポーララインに沿った動きであるとき、エピポーラ制約を満たすという。一方で、図中の歩行者のように交差移動する物体は、このエピポーララインに沿った動きにはならず、エピポーラ制約を満たさない。よって、各画素のオプティカルフローが、エピポーラ制約を満たすか否かを確認することで、移動物体を検出できる。オプティカルフローとは、画像上の各点の動きを二次元ベクトルで表したものである。自転の動きが直進だけでなく、旋回、ピッチ、ロールの回転を伴う場合は、エピポーラ制約を満たすか否かの確認はより複雑になる。そこで、後述する、正規化処理により、回転運動の影響を、画像を変形することでキャンセルし、Fig. 1で示す場合と同様に、移動物体を検知できるようにした。

移動物体が自転車と並走している場合、移動物体であっても、エピポーララインに沿った動きが撮影されるため、エピポーラ制約による検知原理は利用できず、別の検知原理によって検知する。並走移動物体検知アルゴリズムにおける、検知原理を表す模式図を、Fig. 2に示す。この図では、右側にステレオカメラ、左側に移動しているカメラを描き、静止物体と移動物体のある、同じシーンを撮影している様子を示している。ステレオカメラでは、三角測量により、静止物体、移動物体を問わず、距離を計測できる。一方、移動しているカメラでは、時刻 t と時刻 $t+1$ で撮影した画像を使って、三角測量により距離を計測すると、静止物体については正しい距離が計測できるが、移動物体については正しい距離が計測できない。それは、カメラが移動する間に、物体が移動して位置が変わってしまうためであ

る。Fig. 2に示すように、ステレオカメラでも、移動するカメラでも静止物体については実線で示すように、同じ距離が計測されるのに対し、移動物体については破線で示すように、異なった距離が計測される。よって、ステレオカメラで計測した距離と、移動するカメラで計測した距離を比較して、異なっていれば、移動物体であると特定できる。ここまでは、ステレオカメラと移動するカメラを別のカメラとして説明したが、ステレオカメラが車両に取り付けられて、移動する状況を考えれば、ステレオカメラの一方のカメラ、たとえば右側カメラを移動するカメラとして、この原理を適用できる。

2.2. セグメンテーションによる輪郭の鮮明化

本研究では、セグメンテーション技術を活用することで、移動物体の輪郭を鮮明に検知することをねらった。交差移動物体検知アルゴリズムでは、セグメンテーションをオプティカルフローの高精度化に活用し、並走移動物体検知ではオプティカルフローの高精度化に加え、ステレオ視差の高精度化にもセグメンテーションを活用した。オプティカルフロー、ステレオ視差とも、セグメンテーションの活用の考え方は同様であり、ここで概要を解説する。背景にある考え方は、Fig. 3(a)に示すような、モデルフィッティングである。図中に、計測値が点で表されている。計測値にはノイズがあり、また、何らかの理由で計測できず、抜け漏れがあるとすると、計測値が、原理的に一次式で表されるモデルに従うことがわかっていれば、図中に示すように、直線をあてはめることができる。計測値そのものではなく、モデルフィッティングの結果である直線を新たな計測値とすることで、ノイズの影響を低減し、抜けている箇所を含めてより妥当な値に補完することができる。この考え方を画像上での計測値にあてはめた模式図を、Fig. 3(b)に示す。画像がセグメンテーションにより、区画(セグメント)に分割されているものとし、塗り潰して示すセグメント内での計測値が、点で示すように、得られているものとする。例えば、計測値が平面のモデルに従うとすると、セグメント内の計測値を用いて、モデルフィッティングすることで、計測点を代表する斜線のハッチングで示す平面を得ることができる。ここまでは、一般的なモデルフィッ

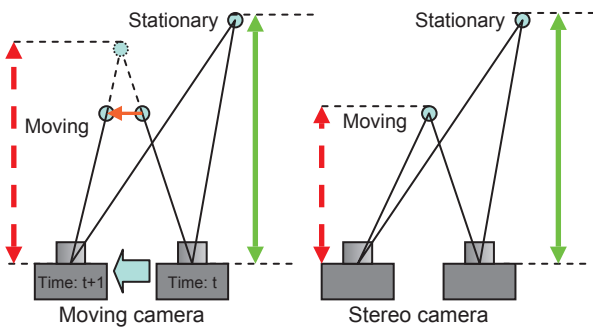
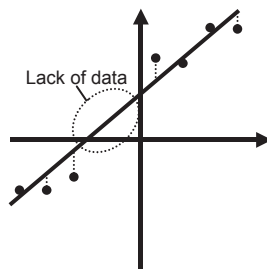
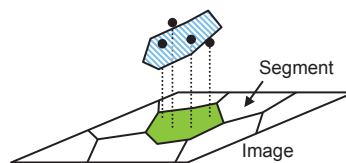


Fig. 2 Principle of detection of moving objects for preceding objects



(a) 1D model fitting



(b) Segmentation-based model fitting on image

Fig. 3 Model fitting based on segmentation

ティングを説明し、どのようなモデルを用いるかは説明しなかったが、本研究では、オプティカルフローについてはモーションモデルを、ステレオ視差については視差モデルを適用した。従来のオプティカルフロー、またはステレオ視差の算出方法では、計算対象の画素の近傍でのテクスチャを利用し、それぞれ、オプティカルフロー、または視差が一様であると仮定して算出する。そのため、テクスチャが乏しい領域では算出できず、また、仮定が成り立たない領域では、特に誤差が大きくなるので、物体の境界を求めることが困難であった。セグメンテーションを用いたモデルフィッティングでは、あらかじめ、オプティカルフローと視差が、モデルに従って一様に変化する領域をセグメンテーションにより求めておくことで、ノイズの影響を低減し、テクスチャの乏しい領域を含めてより妥当な値に補完でき、物体の境界でのオプティカルフローと視差の変化が鮮明に求められるようになった。

2.3. 交差移動物体検知

交差移動物体検知アルゴリズムを実現するために必要な要素は、カメラ運動推定^{(5), (6)}と、オプティカルフロー推定である。カメラ運動推定は、後述する正規化処理のために、必要である。物体の輪郭が鮮明で、ちゅう密なオプティカルフローを求めるため、カラーセグメンテーションに基づくオプティカルフローの算出方法を開発した。カラーセグメンテーションは、イメージセグメンテーションの一手法であり、色の類似性に基づいて画像を分割する手法である。アルゴリズムの全体構成を、Fig. 4に示す。2枚の連続するフレームの画像が、入力として与えられる。「Ego-motion estimation」モジュールでカメラの運動を推定し、カメラの回転の影響を画像上でキャンセルする、正規化処理（「Rectification」モジュール）をおこなう。さらに、「Optical flow computation」モジュールでオプティカルフローの初期値を計算し、最初の画像に対してカラーセグメンテーション（「Color segmentation」モジュール）をおこなう。次に、「Motion model fitting」モジュールで、カラーセグメンテーションによって得られた各セグメントの画像上での動きを表すモーションモデルパラメータを、初期オプティカルフローから、モデルフィッティングによって求める。「Moving region detection」モジュールで、オプティカルフローがエッジポララインに沿っていない画素を特定することで移動物体領域を検知する。主要なアルゴリズム要素

を、以下に詳しく説明する。

2.3.1. 正規化処理

Fig. 4に示す処理フローにおいて、「Ego-motion estimation」モジュールで、カメラの運動パラメータが得られた後、カメラの回転の影響をキャンセルする、正規化処理（「Rectification」モジュール）を説明する。

オプティカルフローは、2枚の連続フレーム画像から求める。カメラ運動推定により、カメラの並進運動と回転運動が求められると、2枚の画像のエッジポール⁽³⁾を画像中心に移し、カメラの回転運動による画像上の動きをキャンセルするように、画像を変形することができる。これにより、Fig. 5に描かれているように、カメラの運動が並進と回転を伴ったものであったとしても、あたかもカメラが並進のみの運動をしたかのような画像に変形できる。この変形により、後段のオプティカルフローが、エッジポララインに沿っているか否か確認する処理が単純になる。つまり、カメラの回転運動の影響がなくなることで、Fig. 1に示す場合のように、容易に移動物体を検知できるようになる。正規化処理をおこなった画像の例を Fig. 6(b)に示す。この例では、Fig. 6(a)の元画像に対して、正規化処理により、画像全体が少し変形されている。

2.3.2. セグメンテーションによるモーションモデルフィッティング

Fig. 4に示す処理フローにおいて、「Optical flow computation」モジュールにおいて算出される初期オプティカルフローと、「Color segmentation」モジュールで得られるセグメントを用いて、「Motion model fitting」モジュールにおいて、モーションモデルパラメータを推定する手法を説明する。

路面、車両、標識など交通環境で見られる物体は、路上を走行する車からの距離で見た場合、平面の集まりとし

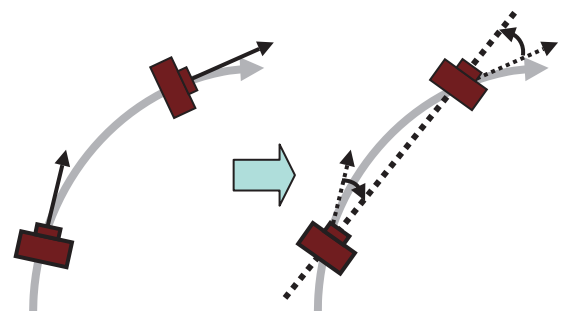


Fig. 5 Concept of rectification

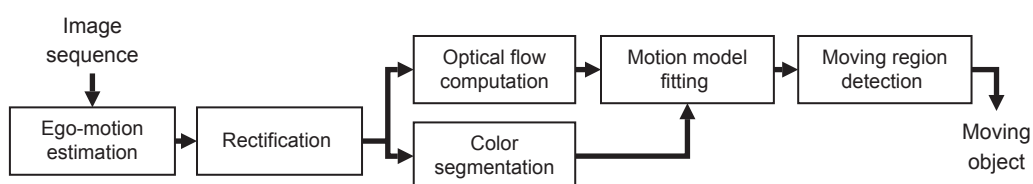


Fig. 4 Block diagram of intersecting object detection

て近似できる。画像上での動きを求める場合、歩行者のような剛体でないものでも、ある程度離れた距離では、体の部分ごとに平面として近似できる。2フレームの画像間での平面の動きは射影変換で表されるので、物体を構成する平面ごとの画像上での動きを、射影変換によってモデル化することは合理的である。そこで、色の類似性に基づいて画像を分割し、分割された各セグメントが平面の一部であり、その動きが射影変換で近似できるものと仮定し、カラーセグメンテーションに基づくオプティカルフロー算出手法を考案した。まず、初期オプティカルフローを、正規化処理後の連続する2フレーム画像を用いて、ブロックマッチング手法により計算する。この初期オプティカルフローは、ブロック内に十分なテクスチャがある領域のみで計算する。さらに、2フレームのうちの最初の画像を、カラーセグメンテーションの一手法である mean-shift 手法⁽⁷⁾により、色の類似性に基づいてセグメントに分割する。Fig. 6(c) はセグメンテーションの例であり、同一色で塗られた画素は、同一のセグメントに属することを表している。

他のセグメンテーションに基づく手法⁽⁸⁾と同様にセグメント境界は物体境界を含んでおり、各セグメントが平面の一部を撮影しているものと仮定し、セグメントごとに初期オプティカルフローを射影変換で表されるモーションモデルにフィッティングすることによって、パラメータを推定する。平面上の点を撮影した場合、あるフレームで画像位置 (x_1, y_1) に撮影された点が次のフレームで (x_2, y_2) に移動したときの動きは次の式でモデル化される。

$$w \begin{pmatrix} x_1 \\ y_1 \\ 1 \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & 1 \end{pmatrix} \begin{pmatrix} x_2 \\ y_2 \\ 1 \end{pmatrix} \quad (1)$$

ここで w は射影変換後のスケールを表す量であり、変換後の第3要素が1になるようにして求められる。 $\{h_{ij}\}$ は、推定すべき射影変換のパラメータである。初期オプティカルフローによって各点 (x_1, y_1) に対応する移動先 (x_2, y_2) が得られるので、それらの対応点を代入することにより、パラメータの推定ができる。誤差の影響を低減するために、実験ではロバスト推定手法の一つである、RANSAC 手法⁽⁹⁾を用いてパラメータを推定した。推定されたパラメータを

使ってセグメント内の各画素の動きを計算すれば、初期オプティカルフローよりも精度の高いオプティカルフローが得られ、カラーセグメンテーションにより鮮明な輪郭が得られる。

2.3.3. 移動物体領域の検知

Fig. 4 に示す処理フローにおいて、「Motion model fitting」モジュールにおいて、精度の高いオプティカルフローが得られた後、移動物体領域を特定する「Moving region detection」モジュールについて解説する。

画像上の移動物体領域は、各画素で、オプティカルフローの方向が、対応するエピポーララインとは異なっていることを確認することで特定できる。また、カメラの回転運動がキャンセルされ、エピポールが画像中心になるように正規化処理をおこなっていることから、ある点に対応するエピポーララインは、単純にその点と画像中心を結ぶ直線として求められる。各画素において、オプティカルフローの向きが、エピポーララインに沿っていない場合に、移動物体であると特定できる。

2.4. 並走移動物体検知

自転車と同一方向に移動する物体は、自律的に移動している点を、画像上の動きから推定される距離とステレオカメラにより計測される距離を比較することによって見つけ出し、移動物体に属するものとして検知する。静止物体では両者は一致するが、移動物体では異なるため、移動物体を特定できる。この手法でもカラーセグメンテーションを利用することで、物体の輪郭が鮮明に求められる。

アルゴリズムの全体をブロックダイアグラムで Fig. 7 に示す。交差移動物体の検知アルゴリズムに対して追加された部分は、セグメンテーションによるステレオ視差算出と動きからの距離推定である。また、画像上での移動物体領域の特定手法は、原理の違いから交差移動物体検知でのものとは異なっている。主要なアルゴリズム要素を、以下に詳しく説明する。

2.4.1. セグメンテーションによるステレオ視差モデルフィッティング

Fig. 7 に示す処理フローにおいて、「Stereo disparity computation」モジュールにおいて算出される初期ステレオ

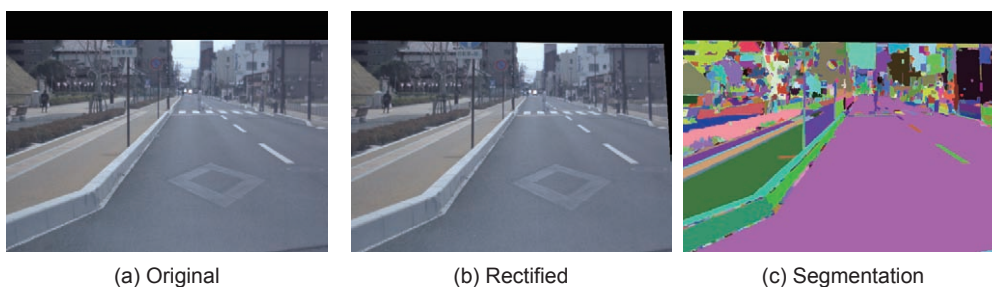


Fig. 6 Examples of rectification and segmentation

視差と、「Color segmentation」モジュールで得られるセグメントを用いて、「Disparity model fitting」モジュールにおいて、視差モデルパラメータを推定する手法を解説する。

オプティカルフロー算出の場合と同様に、初期ステレオ視差を用い、モデルフィッティングによって各セグメントの視差モデルパラメータを推定する。初期ステレオ視差は、ブロックマッチング手法によって求めるため、テクスチャが十分にある領域でしか算出できない。また、遮蔽による誤差を低減するために、基準画像を取り替えて左画像に対する右画像の視差、右画像に対する左画像の視差を求め、両者が一致する場合のみを視差値として採用する処理をおこなう。ステレオの一方の画像を mean-shift 手法により色の類似性に基づいてセグメントに分割し、視差モデルパラメータを、各セグメントに初期視差マップを当てはめることによって推定する。このとき、ステレオカメラの画像は、並行ステレオとなるように補正する。 $\alpha X + \beta Y + \gamma Z = 1$ で表される三次元空間 (X, Y, Z) の平面を撮影したときの、左右カメラの画像上での対応点 (x_1, y_1) , (x_2, y_2) は並行ステレオの場合、次式を満たす。

$$\begin{pmatrix} x_1 \\ y_1 \\ 1 \end{pmatrix} = \begin{pmatrix} \alpha' & \beta' & \gamma' \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_2 \\ y_2 \\ 1 \end{pmatrix} \quad (2)$$

ここで α' , β' , γ' は、次のように定義される。

$$\alpha' = B\alpha + 1 \quad (3)$$

$$\beta' = \frac{f_x}{f_y} B\beta \quad (4)$$

$$\gamma' = f_x B \left(\gamma - \frac{\alpha c_x}{f_x} - \frac{\beta c_y}{f_y} \right) \quad (5)$$

B はステレオカメラの基線長、 f_x と f_y は画素単位での焦点距離の x と y 成分である。 (C_x, C_y) は画像平面と光軸の交点である、主点を表す。 (x_1, y_1) , (x_2, y_2) は初期ステレオ視差を算出した際に得られるので、視差モデルのパラメータ α' , β' , γ' を、式 (2) を用いてセグメントごとに求めることが可能である。得られたモデルパラメータを用いて、各画素の視差を算出することで、初期ステレオ視差よりも精度の高い視差マップが得られ、視差の境界も鮮明になる。

2.4.2. 動きによる距離推定

Fig. 7 に示す処理フローにおいて、「Motion model fitting」モジュールにおいて、精度の高いオプティカルフローが得られた後、「Motion disparity computation」モジュールで動きから距離を推定する処理を解説する。

交差移動物体の検知とは異なり、ステレオによる距離と比較するために、画像上の動きから距離を推定する必要がある。カメラの回転運動がキャンセルされるように正規化されているので、距離 Z は次のように推定できる。

$$Z = T \frac{x}{v_x} = T \frac{y}{v_y} \quad (6)$$

ここで T はカメラの並進運動量であり、未知のパラメータである。 (v_x, v_y) は、点 (x, y) でのオプティカルフローを表している。式 (6) から、距離 Z はオプティカルフローの x または y 成分を用いて求められるが、スケールファクター T の不定性がある。Fig. 8 は、オプティカルフローの x と y 成

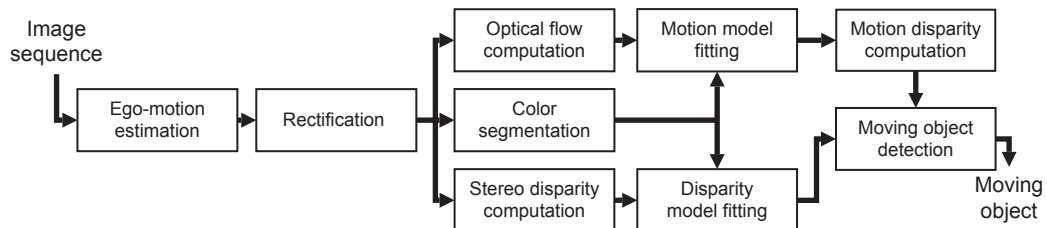


Fig. 7 Block diagram of preceding object detection

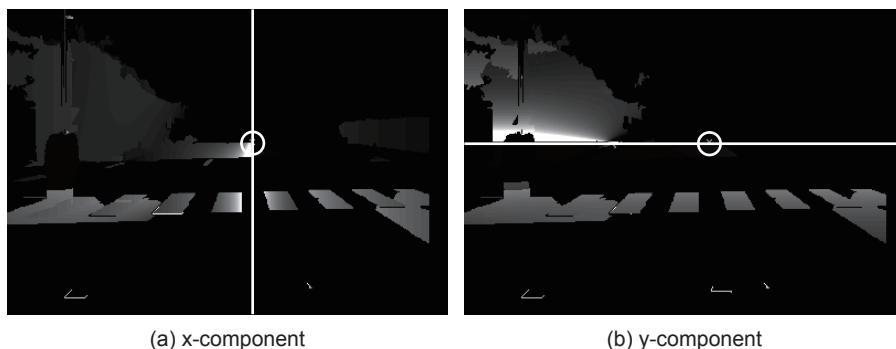


Fig. 8 Intermediate motion disparity computation results

分それぞれから距離を推定した例である。この図では、距離の変化を濃淡で表しており、白に近いほど近距離を、黒に近いほど遠距離を表す。ただし、真黒は距離が算出できなかった領域を表す。どちらの成分からも距離推定が可能ではあるものの、 x 成分を用いた場合では、図中に線で示したFoEを通る垂直線上に、 y 成分を用いた場合では水平方向にそれぞれ大きな誤差が発生する。図中で、丸印はFoEの位置を表している。この誤差はオプティカルフローの各成分が小さいために発生する。この誤差を低減するために、オプティカルフローの方位と大きさを用いることもできる。しかし、簡便さのために、この誤差が画像の特定の領域で発生していることを利用して、各成分での誤差が大きくなる領域の重みを小さくするようにして、二つの距離の重み付き平均を取ることで誤差を低減する。つまり、画像上の点 (x,y) での距離 $Z(x,y)$ を、次のように求める。

$$Z(x,y) = w_x(x,y)Z_x(x,y) + w_y(x,y)Z_y(x,y) \quad (7)$$

ここで $w_x(x,y)$ と $w_y(x,y)$ は画像上の位置 (x,y) での重みで、 $w_x(x,y) + w_y(x,y) = 1$ を満たすように設定した。 $Z_x(x,y)$ と $Z_y(x,y)$ は、 (x,y) でそれぞれオプティカルフローの x および y 成分から推定された距離である。重みの分布と重み付き平均によって得られた距離マップを、Fig. 9 に示す。Fig. 8 に示す各成分による距離マップと比べ、FoEを通る垂直、水平線上で発生していた大きな誤差は、Fig. 9 に示す距離マップでは低減されている。

2.4.3. 移動物体領域の特定

Fig. 7 に示す処理フローにおいて、「Motion disparity computation」モジュールで得られた、動きによる距離と、「Disparity model fitting」モジュールで得られた、精度の高いステレオによる距離を用いて、「Moving object detection」モジュールにおいて、移動物体領域を特定する処理を解説する。

動きからの距離推定では、スケールが不定であるので、直接ステレオによる距離と比較できない。そこで、シーン中の大部分は静止物体であると仮定して、次のように未定のスケール T を推定する。

$$T = \text{median}_{(x,y)} \left(\frac{D_s(x,y)}{D_m(x,y)} \right) \quad (8)$$

ここで $D_s(x,y)$ と $D_m(x,y)$ は画像上の位置 (x,y) での、それぞれステレオによる距離と動きによる距離を表す。メディアンは D_s と D_m が有効なすべての点を用いて計算する。そして、次のような条件を満たす画素を特定することによって、移動物体領域を検知できる。

$$|D_s - T \cdot D_m| > \theta \quad (9)$$

ここで θ は、正のしきい値である。

3. 実験

本手法では、画像分割技術を用いることで、従来のブロックマッチング手法に比べ、オプティカルフロー推定とステレオ視差推定の性能が向上した。これにより、移動物体検知において物体の輪郭を検知することができた。本章では、セグメンテーションによるオプティカルフロー推定とステレオ視差推定の性能についての実験結果を示した後、移動物体検知結果を示す。

3.1. セグメンテーションによるオプティカルフロー推定精度

セグメンテーションによるオプティカルフロー推定精度が、初期オプティカルフローよりも向上していることを定量的に評価した。ここで、セグメンテーションによるオプティカルフローは、前述のモーションモデルフィッティ

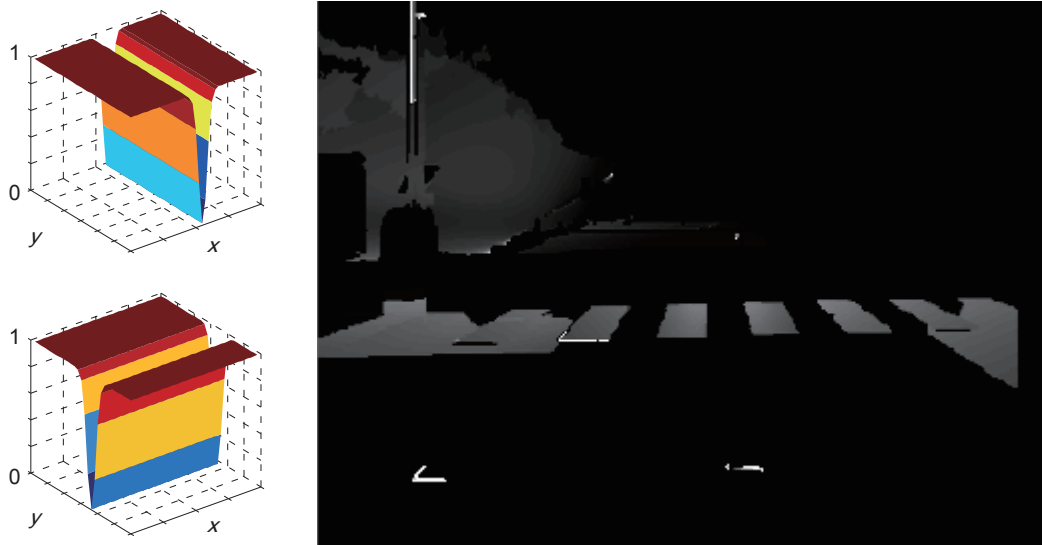


Fig. 9 Weight maps and motion disparity computation result

Table 1 Error in segmentation-based optical flow estimation

Average angular error [deg]	Sequence No.1	Sequence No.2	Sequence No.3
Initial optical flow (Block matching)	11.25 (112330)	15.57 (54162)	10.84 (81210)
Segmentation-based optical flow	8.64 (219010)	9.32 (153671)	4.60 (176364)

Average end-point error [pixel]	Sequence No.1	Sequence No.2	Sequence No.3
Initial optical flow (Block matching)	0.17 (112330)	0.22 (54162)	0.17 (81210)
Segmentation-based optical flow	0.13 (219010)	0.14 (153671)	0.07 (176364)

ングによって得られたモデルパラメータを用いて、各画素のオプティカルフローを算出したものである。Middlebury データセット⁽¹⁰⁾を使って、各時系列画像について求めたブロックマッチングによる初期オプティカルフローと、セグメンテーションによるオプティカルフローについて角度誤差とエンドポイント誤差を、Table 1 に示す。エンドポイント誤差とは、オプティカルフローを表す、ベクトルの長さの誤差である。Table 1 の数値は、角度誤差とエンドポイント誤差について、それぞれ、度と画素の単位で示している。ブロックマッチングではテクスチャレスの場合に、セグメンテーションに基づく手法では、モデルフィッティング誤差が大きい場合に有効なオプティカルフローが得られない。各時系列画像について、有効なオプティカルフローが得られた画素数を、Table 1 のカッコ内の数字で示す。Table 1 に示した誤差値は、カッコ内に示すオプティカルフローが得られた画素数での平均である。表の誤差値から明らかなように、セグメンテーションに基づくオプティカ

ルフロー推定により、従来のブロックマッチングによる手法より高精度でオプティカルフローを推定できた。また、有効な画素数が多いことは、テクスチャレス領域でオプティカルフローを補完できたことを示している。つまり、Fig. 4 と Fig. 7 の「Optical flow computation」モジュールで算出される初期オプティカルフローが、同図の「Motion Model Fitting」モジュールでの処理後に精度向上したことが確認できた。このセグメンテーションを用いたオプティカルフロー推定により、動きの境界が鮮明になったことで、物体の輪郭を鮮明に求められるようになった。

3.2. セグメンテーションによるステレオ視差推定精度

Fig. 10 は、ブロックマッチングによる初期ステレオ視差マップと、セグメンテーションによるステレオ視差マップの計算例を示している。ここで、セグメンテーションによるステレオ視差は、前述の視差モデルフィッティングによって得られたモデルパラメータを用いて、各画素の視差

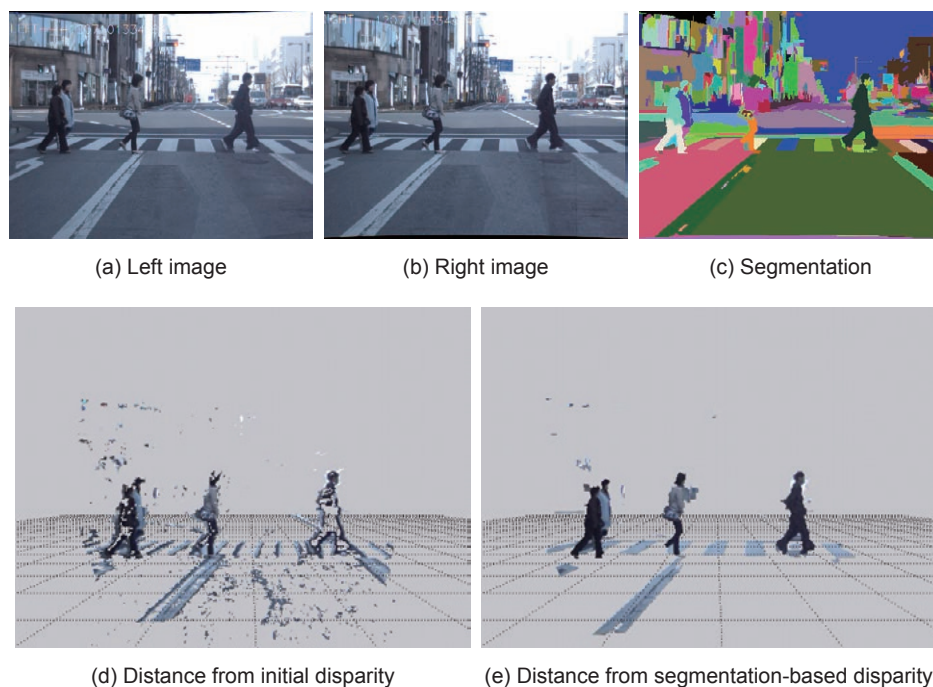


Fig. 10 Comparison of initial and segmentation-based disparity

を算出したものである。(a), (b) はステレオカメラの左右画像であり, (c) は右画像をカラーセグメンテーションした結果である。(d) にブロックマッチングによる初期ステレオ視差を, (e) にセグメンテーションによるステレオ視差推定結果を示す。初期ステレオ視差ではノイズが多く, 特に, 図中の歩行者の胴体部分のように, ブロックマッチングでは, テクスチャレス領域において, 視差が計測できない。セグメンテーションに基づく手法では, 視差モデルを用いることによってテクスチャレス領域で視差を補完でき, またノイズの影響が低減したことで輪郭が鮮明になった。つまり, Fig. 7 における「Stereo disparity computation」モジュールにより算出された初期ステレオ視差が, 同図の「Disparity model fitting」モジュールでの処理後に精度向上したことが確認できた。ステレオ視差の場合のモデルは, パラメータが少なくフィッティングが容易であることか

ら, オプティカルフローのように定量的な評価はおこなわなかったが, 他のシーンでも, Fig. 10 と同様に, セグメンテーションに基づく手法で, 従来のブロックマッチングに比べ良好な距離推定結果が得られた。

3.3. 交差移動物体検知実験

一般道路で収集した画像を用いて, 前述の交差移動物体検知手法を試みた。横断歩道を横断している歩行者に向かって走行しているシーンを実験に用いたが, 十分な距離を取り, 通常の安全な走行条件で撮影した。歩行者の検知結果を Fig. 11 に示す。この図では, 撮影した動画の 1 フレームごとの検知結果を示している。撮影した画像を図の (a) に示しており, 最初のフレームで歩行者が自転車線に進入し始め, 最後のフレームで自転車線の中央に到達した。図の (b) で塗り潰した領域が移動物体として検知された領域

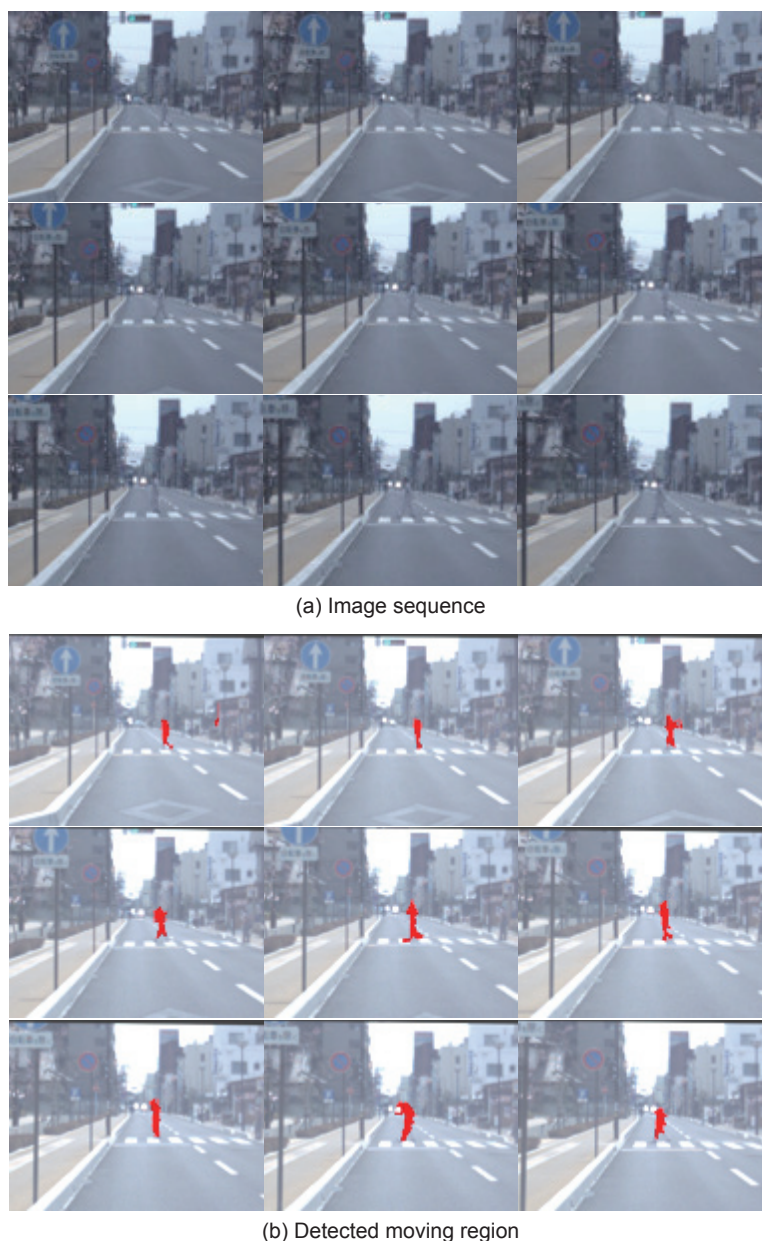


Fig. 11 Detection results for intersecting object

である。大きな誤検知もなく、歩行者を良く検知できており、輪郭も概ね正しく検知できた。この実験では簡便さのために、移動物体領域を特定する際、水平方向の動きのみから判定をおこなっており、検知できる移動物体は画像上でFoEに向かって移動する物体に限られる。また、誤検知を低減するために、大きさや画像上の位置などから、移動物体か否かを判断する後処理を加えた。

3.4. 並走移動物体検知実験

次に、並走する移動物体を検知する手法を、実際に高速道路で撮影したシーンに適用した。この実験で用いたシーンでは、自車の隣車線に並走する追い越し車両があり、また自車の前方には別の車両が走行していた。記録した動画から、8フレームごとの画像を、Fig. 12(a)に示す。

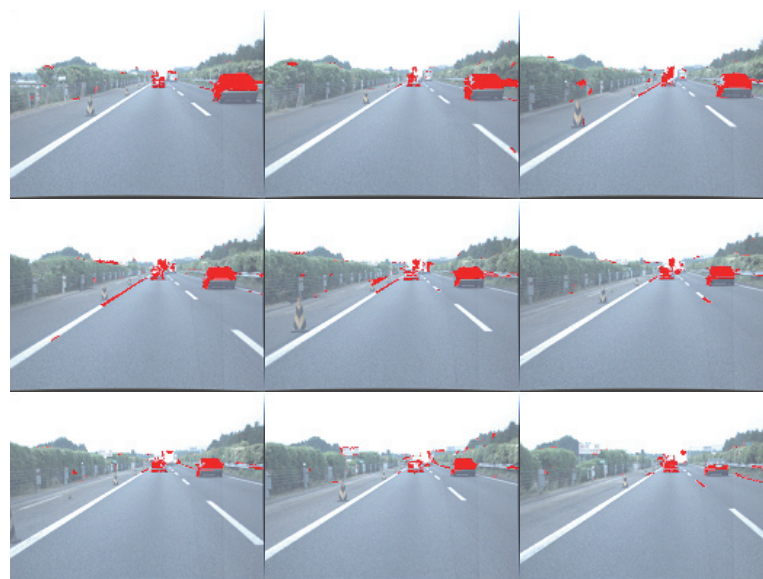
Fig. 12(b)には、移動物体として検知された領域を塗り潰して示す。先行車と隣車線の車両の両方を本手法により検知できた。また、路面に近い影の部分を除き、追い越し車両の輪郭を検知できた。この実験ではフレームごと、画素ごとに移動物体か否かを判定しており、誤検知を低減する後処理は用いなかった。誤検知が多少見られるが、断続的に発生する誤検知であるので、クラスタリングやトラッキングなどの後処理を加えることによって、低減できる。

4. 結論

車両に取り付けたカメラで撮影した動画から自律移動物体を検知する二つの手法を開発した。第1のアルゴリズムは、単一のカメラを用いる手法で、横断歩行者や交差移動



(a) Image sequence



(b) Detected moving region

Fig. 12 Detection results for preceding object

する車両のように、カメラと検知対象が別の方向に移動する場合に適用できる。第2のアルゴリズムは、ステレオカメラを用いた手法である。システムはより複雑になるが、ステレオによる距離と動きからの距離を比較することで、並走する移動物体を含む、より一般的な移動物体を検知できる。実験により、本手法を用いて移動物体の鮮明な輪郭が得られること、およびその結果、移動物体を検知できることを確認した。

現状のプログラムは、1フレームの処理に数十秒の計算時間を要している。この処理時間の短縮が今後の課題である。また、カラーセグメンテーションをオプティカルフローやステレオ視差の算出に用いる際に、セグメント境界が物体の境界を含むという仮定をしているが、この仮定が成り立たない場合に誤検知や不検知を発生しやすくなることがある。この解決も、今後の課題である。

参考文献

- (1) Agrawal, M., Konolige, K., Iocchi, L.: Real-time detection of independent motion using stereo, IEEE workshop on Motion and Video Computing (WACV/MOTIONS)(2005)
- (2) Vaudrey, T., Wedel, A., Rabe, C., Klappstein, J., Klette, R.: Evaluation of moving object segmentation comparing 6d-vision and monocular motion constraints, IVCNZ08, p.1-6(2008)
- (3) Hartley, R. I., Zisserman, A.: Multiple View Geometry in Computer Vision, Second Edition, Cambridge University Press(2004)
- (4) Ogale, A. S., Fermüller, C., Aloimonos, Y.: Detecting independent 3d movement, Handbook of Geometric Computing Applications in Pattern Recognition, Computer Vision, Neural computing, and Robotics, E. Bayro-Corrochano, Springer Verlag(2005)
- (5) Hartley, R. I.: In defense of the eight-point algorithm, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.19, No.6, p.580-593(1997)
- (6) Domke, J., Aloimonos, Y.: A probabilistic framework for correspondence and egomotion, ICCV Workshop on Dynamic Vision(2005)
- (7) Comaniciu, D., Meer, P.: Mean shift: A robust approach toward feature space analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.24, No.5, p.603-619(2002)
- (8) Bleyer, M., Gelautz, M., Rhemann, C.: Colour segmentation-based computation of dense optical flow with application to video object segmentation, ÖGAI-Journal, Vol.24, No.1, p.11-15(2005)

- (9) Fischler, M. A., Bolles, R. C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography, Commun. ACM, Vol.24, No.6, p.381-395(1981)
- (10) Baker, S., Scharstein, D., Lewis, J., Roth, S., Black, M. J., Szeliski, R.: A database and evaluation methodology for optical flow, ICCV2007(2007)

■ 著 者 ■



西垣守道



Yiannis ALOIMONOS