

移動するカメラの動画像に特化した オプティカルフローの計算手法

— 二輪車の自律走行のための視覚処理 —

興梠 正克 村岡 洋一

早稲田大学 理工学部

{kourogi, muraoka}@muraoka.info.waseda.ac.jp

梗概 車載カメラから撮影した画像に基づいて乗り物を制御する応用 [3] [4] が、近年になって注目されている。本稿では、移動するカメラが撮像する動画像に特化したオプティカルフローの計算により、そのカメラの自己運動の推定と接近する障害物の検出を行なう手法について述べる。動画像中に現れるオプティカルフロー場は、回転運動、並進運動および移動物体による運動の3つの成分がそれぞれ引き起こした場の重ね合わせである。本手法は、画像全域に遍在してその方向が既知である回転運動が引き起こすフローの成分を分けて計算する。そのための方法として、垂直方向と水平方向に沿った一次元オプティカルフロー計算を利用する。これにより計算された成分から、回転運動の推定を行なう。この推定結果から回転運動が引き起こすオプティカルフロー成分を補償する。次に FOE を中心として広がる方向への一次元オプティカルフロー計算を行なう。並進運動による成分は、近接している障害物の位置に強く現れるので、これに基づいて障害物を検出することができる。製作した二輪車に対して、本手法に基づいて自律走行させる実験を行ない、転倒の防止と障害物の検出ができることを確認した。

Computation of optical flow for image sequences from a moving observer

— Visual processing for an autonomous bicycle —

Masakatsu Kourogi Yoichi Muraoka

School of Science and Engineering, Waseda University
3-4-1 Ohkubo Shinjuku-ku, Tokyo 169, JAPAN.

Abstract In this paper, we propose a method for estimating egomotion and detecting proximate obstacles. This method specializes in an image sequence taken from a moving observer. An optical flow field appearing in an image sequence can be regarded as a superposition of fields caused by rotation and translation of egomotion and moving object. The method attempts to compute each field separately by means of one dimensional optical flow (1-D optical flow) computation. First, it computes the component caused by rotation of egomotion, and statistically estimates the rotation. Second, it compensates for the optical flow field caused by the estimated rotation and computes the component caused by translation, which is expansion from FOE. We conducted experiments on a bicycle (1/24 model, radio-controlled) and confirmed that it can move around without falling down and avoid obstacles.

1 はじめに

ロボットの自律的な移動のための主要な知覚手段として、視覚センサーが注目されている。周囲の環境と自己の運動状態に関する手がかりを提供するから

である。自律ロボットの多くの従来研究では、視覚情報に基づいた3、4輪構成のロボットの自律的走行が試みられてきた。しかしながら、転倒したり横滑りすることのある、動力学的に不安定な対象を視覚センサーによって制御しようとする試みはほとんどな

れていない。このような対象の制御には、ジャイロセンサーや傾斜センサーが主として利用され [1][2]、視覚センサーは使われることがなかった。

本研究では、自由度が大きく動きの不安定な対象である二輪車を視覚フィードバックのみにより自律走行させることを目標とする。二輪車の自律走行には、(a) 自己運動の推定と (b) 近接する障害物の検出の二つを実時間で実現する必要がある。

オプティカルフローは、理論的にはこれらの課題を解決できる情報を提供できる。しかしながら、信頼できるオプティカルフローを計算することはそもそも難しい問題であり、その解決には高い計算コストが必要とされるためその実時間処理は難しい。

そこで本研究では、オプティカルフローを引き起こす原因である回転運動、並進運動によるオプティカルフロー成分を別々に計算する方針を取る。回転運動の3成分によるオプティカルフローと、並進運動によるオプティカルフローはそれぞれその方向が既知であるので、成分ごとを算出する方法として、良設定で計算コストの小さい次元オプティカルフロー計算を利用する。

本手法に基づく視覚フィードバック制御を、製作した二輪車に対して適用して、これを自律走行させる実験を行なった。その結果、フィードバック制御なしでは転倒してしまう状況下においても、その転倒を防ぐことができることを確認した。また、いくつかの障害物配置されている環境中を、それらとの衝突を回避して走行できることを確認した。

本論文の構成は以下の通りである。2章では、オプティカルフローに基づいた視覚フィードバックの実現を難しくしていた問題点を明らかにして、その解決策について述べる。3章では、本手法が用いる次元オプティカルフロー計算について述べ、それが利用可能となる条件を明らかにする。4章では、本手法による自己運動の推定、近接する障害物の検出を行なうアルゴリズムについて述べる。5章では、このアルゴリズムによって実現できる、二輪車の自律走行に不可欠な視覚フィードバック制御について述べる。6章では、製作した二輪車のラジコンに対してこの視覚フィードバックを応用する。この二輪車の走行実験を行ない、本手法の有効性を検証する。最後に、7章では本稿で述べたことの要約を行なう。

2 問題の分析とその解決策

二輪車を自律走行させるためには、(a) 自己運動の推定と (b) 近接する障害物の検出の二つを実現する必要がある。二輪車は動力学的に不安定な制御対象であるため、自己運動の状態を常に監視してフィードバックをかけなくてはならず、(a) が必要となる。また、障害物との衝突を回避するために、(b) が必要となる。

多くの従来研究では、画像情報に基づいてこれらの二つを実現する方法として、オプティカルフローの利用を有視視してきた。オプティカルフローにはそのために必要な情報が含まれることが、理論的に示されているからである [10]。

しかしながら、精度の高いオプティカルフローの計算はそもそも難しく、またその計算コストが高いため実時間処理が難しいという問題点があった。このためオプティカルフローに基づいた視覚フィードバックの実現は難しかった。本研究では、この問題はオプティカルフローの計算のされ方に起因していると分析する。

カメラの座標系を図1のように定義すると、そのカ

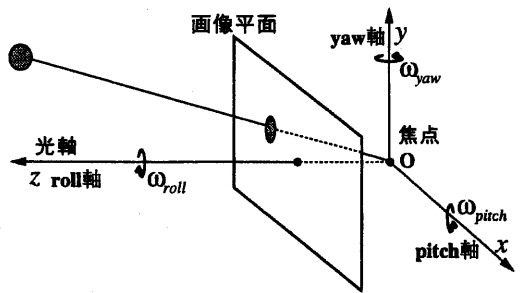


図1: カメラの座標系

メラが運動 (回転運動+並進運動) するときその画像中に現れるオプティカルフロー場は、以下の図2に示す各成分の重ね合わせとして表現できる。ほとんどの従来研究では、この重ね合わせの結果であるオプティカルフロー場を最初に計算して、その後それぞれ成分への分解を試みていた。このやり方では、フロー場の計算とそのフローの各成分への分解という難しい問題を二つも解く必要がある。さらに、計算すべきオプティカルフローはすべての成分の重ね合わせであるため、個々の成分に関する部分的に予測が得られていても、それを活用することができなかつ

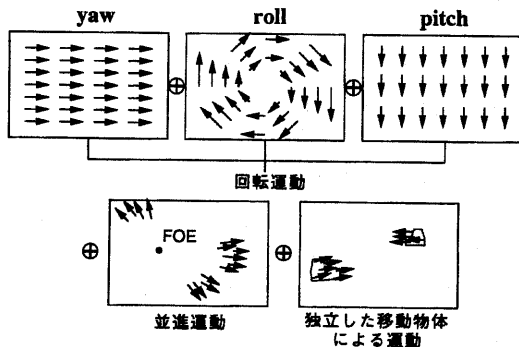


図 2: オプティカルフロー場の成分

た。例えば、カメラの yaw 軸周りの回転運動に関する予測だけが与えられているとしても、オプティカルフローの計算過程にとって、それは手がかりとして利用できない。したがって、このやり方はあらゆる方向と大きさのオプティカルフローを、いかなる予測も使うことなしに計算できる、いわば万能的な計算手法を必要とする。これが、オプティカルフローのロボットビジョンへの応用を難しくしていた。

ここでは、そのような計算手法を追求することはせずに¹、ロボットビジョンや乗り物の自動操縦などの応用 (eg. ITS) に目的を限定して、移動するカメラが撮像した動画像に特化した手法を考える。

そこで本研究では、オプティカルフローを成分ごとに分けて計算することにより、この問題の解決を試みる。回転運動による 3 つのオプティカルフローの成分 (yaw, roll, pitch 軸周り) は、それぞれ画像全域に遍在しており、そのフローの方向が既知である。この方向に沿った一次元オプティカルフローを計算することで、それぞれの成分を計算することを考える。そして、画像中の数点におけるオプティカルフローの成分から、その運動の推定が可能である。

また、並進運動による成分は、広がりを中心 FOE (Focus of Expansion) の位置が分かっている²、そのオプティカルフロー成分の方向は既知となる。回転運動の推定と同様に、一次元オプティカルフロー計算により、並進運動の成分を求めるを試みる。この成分の大きさは物体までの距離に反比例しているため、衝突する危険性のある障害物の検出に利用

¹ 静止したカメラから粒子の流動の速度、方向を計測するような応用例ではオプティカルフローの万能な計算手法が必要である

² 直進運動しか取らない二輪車では、画像中心が FOE となる

することができる。

3 一次元オプティカルフロー

前章では、一次元オプティカルフロー計算によってフローの各成分を求めることで、カメラの回転運動の推定と障害物の検出を行なうアプローチについて述べた。本章では、その一次元オプティカルフロー計算が正当化される条件について、明らかにする。

3.1 一次元オプティカルフロー計算

一次元オプティカルフローは、動画像を特定方向に沿って着目したときの見かけ上の移動ベクトルとして定義できる。画像中の点 (x, y) の時刻 t における明るさを $E(x, y, t)$ とする。x 軸方向に沿った一次元オプティカルフローを u_0 とおけば、以下の等式が成り立つ。

$$E(x + u_0 \delta t, y, t + \delta t) = E(x, y, t) \quad (1)$$

式 1 の左辺をテイラー展開して極限 $\delta t \rightarrow 0$ を取ると、一次元オプティカルフローの制約方程式として以下の式が得られる。

$$\frac{\partial E}{\partial x} u_0 + \frac{\partial E}{\partial t} = 0 \quad (2)$$

したがって、特定の方向に沿って、その一次元オプ

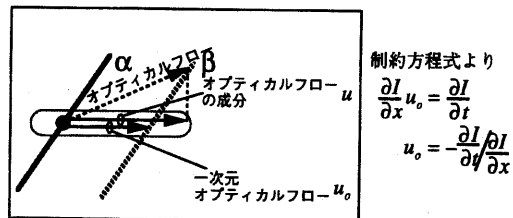


図 3: 一次元オプティカルフローとオプティカルフローの成分

ティカルフローを計算することは良設定な問題であり、小さな計算コストで実現できる。しかしながら、良く知られた二次元オプティカルフローの制約方程式は、フローを (u, v) とおくと、以下の通りである。

$$\frac{\partial E}{\partial x} u + \frac{\partial E}{\partial y} v + \frac{\partial E}{\partial t} = 0 \quad (3)$$

このため、一次元オプティカルフロー u_0 とその方向へのオプティカルフローの成分 u は一般的には一致

しない。式2と式3を比較すると、 $u_o = u$ となるためには、 $\partial E / \partial y = 0$ または $v = 0$ のいずれかが満たされている必要があることが分かる。すなわち、以下の二つの条件のいずれかが満たされていなければならない。

条件1 着目する方向に直交する画像勾配が0である

条件2 着目する方向に直交するオプティカルフローの成分が0である。

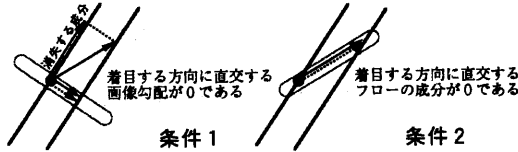


図4: 一次元オプティカルフロー計算が妥当となる条件

3.2 予測値を利用した計算方法

式1に基づく、画像勾配からの一次元オプティカルフローの計算方法には限界がある。それは、離散化された連続画像においては、大きな移動量を持つオプティカルフローを計算することはできないことである。そのようなフローは、連続性を仮定して導かれた制約方程式2をもはや満たさなくなるからである。

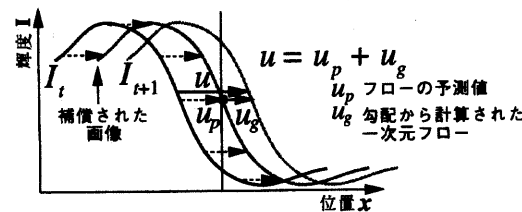


図5: 予測付きの一次元オプティカルフロー計算

この問題に対処するために、ここでは一次元オプティカルフローの予測値を利用した計算方法を考える。図5が示すように、予測値 u_p が与えられているとき、画像をこの移動量だけ補償する。この補償によって、勾配法によって計算すべきフロー u_g を小さくできる。この予測値を用いた計算方法により、単純

な勾配法では計算できない大きな移動量のフローを求めることが可能となる。

4 アルゴリズム

4.1 画素データの選別

特定の方向に沿った一次元オプティカルフロー計算によって、その方向へのオプティカルフローの成分が得られるには、3章で述べられた条件1より、その方向に直交する画像勾配が0でなければならない。理想的なエッジは、このような性質を持つ画像要素であるが、実画像中にはそのようなエッジはほとんど存在しない。ここでは、条件1を、その勾配の絶対値が一定のしきい値よりも小さいという条件に緩和する。

条件1' 直交する方向への画像勾配の絶対値が、しきい値 θ_g よりも小さい

4.2 回転運動推定の反復計算アルゴリズム

図2に示されるように、カメラの回転運動が引き起こすオプティカルフローの垂直成分は pitch 軸周りと roll 軸周りの回転運動による成分の和であり、水平成分は yaw 軸周りと roll 軸周りの回転運動による成分の和である。したがって、並進運動および移動物体による運動の影響を受けていない画素上では、そのオプティカルフローの垂直成分と水平成分から回転運動を推定できる。ここでは、多くの画素上において計算されたオプティカルフローの成分を統計的に処理して、カメラの回転運動を推定することを考える。これを以下のようなアルゴリズムによって行なう。

まず、垂直成分と水平成分を計算するのに適する(すなわち前述の条件1'を満たす)画素をそれぞれ選別して、その画素上での一次元オプティカルフロー計算によりそれぞれの成分を求める。計算された水平成分からは、yaw 軸周りと roll 軸周りの回転運動 $\omega_{yaw}, \omega_{roll}$ をそれぞれ推定して、垂直成分からは pitch 軸周りの回転運動 ω_{pitch} を推定する。この回転運動の推定値は、条件1'を満たすごく少数の画素の上で計算されたオプティカルフローの成分に基づいているため、その信頼性は低い。この推定値の精度を向上させるために、以下のような反復計算を行なう。

推定された回転運動によって引き起こされるオプティカルフロー場を計算して、画像中の各点におけるその水平成分 u_p と垂直成分 v_p をそれぞれ求める。これらの成分を使って、一次元オプティカルフローの

再計算を試みる。水平成分の再計算には、3章で述べた条件2を満足させるために、直交する方向への成分である v_p を画像から前もって補償しておく。それと同時に u_p を一次元オプティカルフロー計算の予測値として利用する。垂直成分についても同様の計算を行なう。この再計算された成分に基づいて回転運動の推定を行なうことで、前回の推定結果よりも精度を向上させることができる。この反復計算を、推定結果が一定の値に収束するまで繰り返して、その収束した値を推測値とする。

この反復アルゴリズムは、以下のようにまとめられる。

1. 回転成分によるオプティカルフローの水平、垂直成分を計算するのに適した画素を選別して、その画素上での一次元オプティカルフロー計算によりそれぞれの成分を計算する。計算された成分より、最小二乗法により回転運動を推定する。
2. 推定された回転運動 (ω_{pitch} , ω_{yaw} , ω_{roll}) が引き起こすオプティカルフロー場を計算して、それぞれの水平成分 u_p と垂直成分 v_p を計算する。
3. 水平成分の再計算を、それに直交する成分である v_p を補償した画像で、 u_p を予測値とした一次元オプティカルフロー計算により行なう。垂直成分の再計算も同様に行なう。計算された成分から最小二乗法により回転運動を推定する。
4. この推定結果が前回の推定結果と変わらなくなるまで、2-3を繰り返す。

4.3 近接する障害物の検出

回転運動によって引き起こされたオプティカルフローの成分を補償する(取り除く)と、画像中に残っているのは並進運動による成分と独立した移動物体による成分である。このうち並進運動が引き起こすオプティカルフローの成分は、FOEを中心とした広がりを持つ。この方向に沿った一次元オプティカルフロー計算により、その成分を求めることができる。並進運動による成分の大きさは、その対象までの距離に反比例する。したがって、この成分が大きな画素が集中している領域を探することで、近接する障害物を検出することができる。

直進運動しか取らない場合では、画像中の位置 (x, y) のフロー成分 (u, v) は、直進速度を W 、その

対象までの距離を Z とすると、以下の式で表される。

$$(u, v) = \left(\frac{xW}{Z}, \frac{yW}{Z} \right) \quad (4)$$

したがって、その対象との衝突までの時間 (time to contact) $t_c = Z/W$ を、並進運動による成分 (u, v) から以下のようにして求めることができる。

$$t_c = \frac{x}{u} = \frac{y}{v} \quad (5)$$

5 二輪車の制御

本章では、4章で述べたアルゴリズムにより得られた情報に基づき、二輪車を転倒させずに、また障害物と衝突することなく走行させる制御方法についてそれぞれ述べる。なお、ここでは二輪車の座標系を以下の図6に示すように定義する。

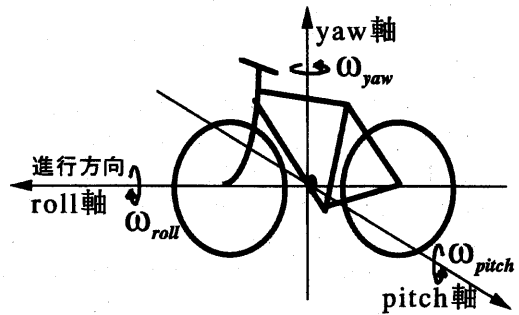


図6: 二輪車の座標系

図1で定義したカメラの座標系と一致するように定義した。

5.1 二輪車の旋回運動

二輪車は、その動力学的な性質により、同じ場所で静止することができない。このため、自律ロボットの従来研究にしばしば見られる「その場で停止して向きを変える」という便利な制御行動を取ることができない。このため、すべての行動を直進運動と旋回運動の組合せだけで実現する必要がある。

しかしながら、二輪車は3、4輪構成の乗り物とは異なり、車輪の方向を単に変更するだけでは旋回しない。高速で直進する二輪車は、いわゆるジャイロ効果により、その進行方向を変えることがほとんどできなくなるからである。

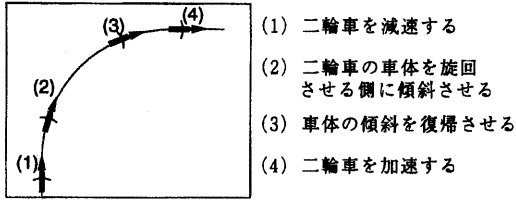


図 7: 二輪車の旋回運動

二輪車を旋回させるには以下の図7に示す手順を踏む必要がある。ところが、二輪車にはその直進速度が減速されると転倒しやすくなる性質がある。すなわち、二輪車が旋回運動している間は、転倒する危険性があるので、これを防止する必要がある。

二輪車の転倒は、初期のごく小さな倒れ込みに始まる。この倒れ込みが補正されないまま放置されると、その速度は時間の経過とともに加速する。やがてその倒れ込みは回復不能な速度に達して、転倒を引き起こす。

したがって、この二輪車の転倒を防ぐには、その始まりの兆候である初期の倒れ込みを検出して、これを早期に補正してやればよい。二輪車の倒れ込みの運動は、図6で定義した座標系でroll軸周りの回転運動 ω_{roll} として表される運動と等価である。つまり、4章で述べたアルゴリズムによって推定される回転運動 ω_{roll} を常に監視して、その運動の推定値が一定値以上になったとき、それを補正するようにフィードバック制御をかけることにより実現可能である。この視覚フィードバックを転倒防止フィードバックと呼ぶことにする。

また、二輪車を直進運動させるフィードバック制御は次のようにして実現可能である。二輪車の方向変化の運動を表しているのはyaw軸周りの回転運動 ω_{yaw} なので、前述のアルゴリズムにより推定される ω_{yaw} を監視して、この値を0に補正するようにフィードバック制御をかける。この視覚フィードバックを直進フィードバックと呼ぶことにする。

5.2 障害物の回避

二輪車が直進しているときは、障害物回避の問題は簡単に解決できる。すなわち、近接する(衝突までの時間 t_c が小さな)物体が正面に存在するときだけ、それを回避する行動を取ればよい。しかしながら、旋回中の二輪車の障害物回避の問題はずっと複雑であ

る。この問題の解決のため、ここでは画像中に検出された近接するそれぞれの物体について、その物体が二輪車の正面に到達する時間 t_f (time to front)を計算する。この時間は、その物体の方向 θ を二輪車の方向旋回の角速度 ω_{yaw} から、 $t_f = \theta / \omega_{yaw}$ として得られる。衝突までの時間 t_c と、正面に到達するまでの時間 t_f が等しいとき、二輪車はその物体と衝突する。旋回中の衝突回避の問題は、この二つの時間を比較して、それらが等しいときに回避行動を取ることで解決できる。

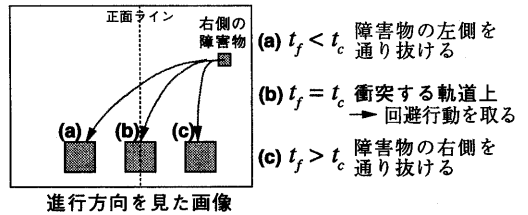


図 8: 衝突する障害物の検出方法

6 実験とその結果

6.1 実装

ここでは、CCDカメラを装備した二輪車のラジコンを製作した。このラジコンは、無線の制御信号により直進速度と車体の傾斜の二つを制御することができる。

4章で述べたアルゴリズムをPC(CPU:Pentium-75MHz/OS:Linux)上に実装した。このアルゴリズムによる平均的な計算コストは、自己運動の回転成分の推定に15ミリ秒、障害物の検出に20ミリ秒であり、おおむね毎秒30フレームの実時間処理を実現できた。また、カメラの運動と障害物の位置が既知である合成画像に対して本手法を適用して、良好な結果をすでに得ている[5]。

6.2 実験とその結果

ここでは、前章で述べた視覚フィードバック制御によって、二輪車が転倒せずに走行できることと、障害物を回避できることを調べる3つの実験を行なった。二輪車が走行する環境はコンクリートの路面上であり、視界内には空を含む遠景が入る。

最初の実験では、その二輪車を転倒しやすくする

ために、片側だけに重りを載せて不均衡な状態にする。この状態で二輪車を走行させて、本手法による転倒防止フィードバックと直進フィードバックの両方をかけた場合と、いかなるフィードバックもかけなかった場合を比較して、本手法の有効性を検証する。図9にそれぞれの場合の二輪車の走行軌跡を示した。フィードバックをかけない走行では、すぐに重りが載せられた側に進路が大きく逸れはじめて、やがて転倒した。これに対して、フィードバックをかけた走行では、進路はいくらか重りの載せられた側に逸れたが、転倒せずに走り続けることに成功した。

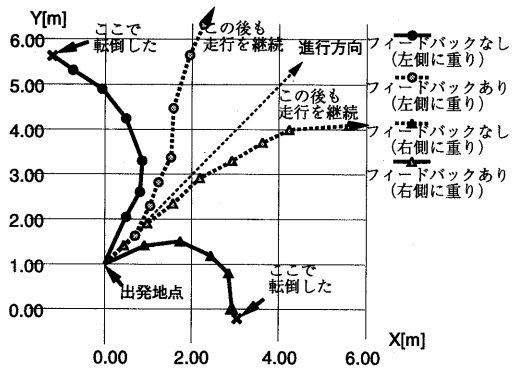


図 9: 実験結果 1

次の実験では、約5度の傾斜のある斜面上を登り上がるように二輪車を走行させることで、不均衡な状態にする。本手法による転倒防止フィードバックおよび直進フィードバックをかけた場合と、かけなかった場合を比較する。図10にそれぞれの場合の二輪車の走行軌跡を示した。フィードバックをかけなかった走行では、二輪車の進路は傾斜している方向に大きく逸れていく、やがて転倒した。これに対して、フィードバックをかけた走行では、進路はやや傾斜の方向に逸れたが、転倒することなく走行を持続することができた。

最後の実験では、走行環境中に4つの障害物(20 cmx20 cm, テクスチャのある直方体の箱)を2 mおきに配置して、二輪車をそのうちの一つの障害物に向けて走行させた。走行には、本手法による転倒防止フィードバックと障害物回避フィードバックをかけた。図11に、その二輪車の走行軌跡を示した。その結果として、多くの場合では障害物との衝突を回避するように走行することができた。

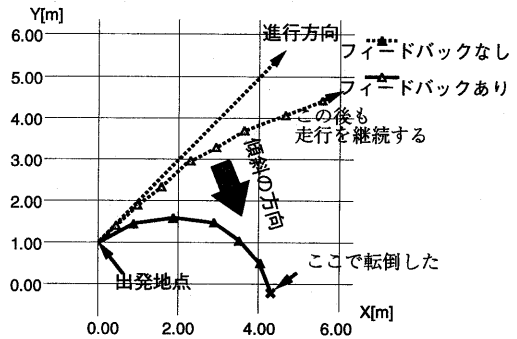


図 10: 実験結果 2

しかしながら、最初の障害物にそのまま真っすぐに衝突する失敗例もいくつか見られた。これは、障害物が画像の中心領域付近に映ってしまう場合、それが作り出すオプティカルフローの成分は小さくなるため、その検出に失敗したことが原因と考えられる。

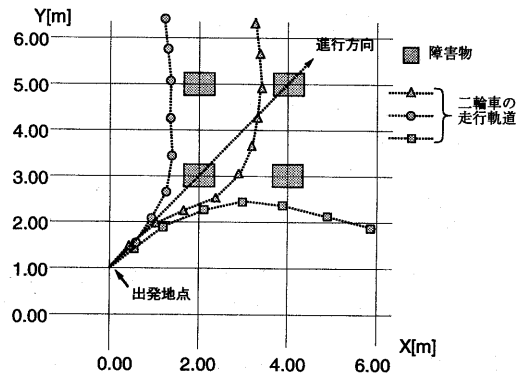


図 11: 実験結果 3

7 まとめと今後の課題

本稿では、移動するカメラから撮像された動画像について、そのオプティカルフローの計算を、回転運動と並進運動による成分にそれぞれ分けて計算するアプローチを提示した。フローを成分ごとに計算する方法として、一次元オプティカルフロー計算を用いた。これによって計算されたフローの成分から、自己運動の推定と近接する障害物の検出を行なう手

法を提案した。本手法は、通常のPC上で特殊なハードウェアなしに実時間(毎秒30フレーム)処理が可能な、計算コストの小さなものである。そして、本手法による自己運動の推定と障害物検出の結果に基づいて、二輪車を制御する視覚フィードバック行動について述べた。この視覚フィードバック制御を、実際に製作した二輪車の自律走行に適用して、その有効性について確認した。

提案された反復計算アルゴリズムは、並進運動および移動物体によるオプティカルフローの成分が、回転運動による成分と比べて著しく大きくなるような動画像に対しては正しく動作しない。今後の課題としては、この手法が正しく働く限界について、明らかにしていきたい。

参考文献

- [1] 盛再権, 山藤和男: 一輪車の姿勢安定と運動制御に関する研究, 日本機械学会論文集(C編)61巻105号: pp.1042-1049. pp.87-95, 1975.
- [2] 山藤和男, 河村隆: 同軸二輪車の姿勢および走行制御に関する研究, 日本機械学会論文集(C編)54巻501号: pp.1114-1121, 1988.
- [3] 伊東他: 走行環境認識のための画像処理手法の検討, 信学技報, PRMU97-25(1997,5), pp.1-8.
- [4] 二宮他: オプティカルフローによる移動物体の検知, 信学技報, PRMU97-28(1997,5), pp.25-32.
- [5] 興侶他: 一次元フロー計算による運動復元と障害物検出, 情処研報, 97-CVIM-105-1(1997,5), pp.1-8.
- [6] R.Sharma, Y.Aloimonos: Early Detection of Independent Motion from Active Control of Normal Image Flow Patterns, IEEE Transactions of Systems, Man and Cybernetics, 26: pp.42-52, 1996.
- [7] C.Fermüller: Navigational Preliminaries, Active Perception: pp.103-150, 1993.
- [8] J.Weber, J.Malik: Robust Computation of Optical Flow in a Multi-Scale Differential Framework, International Journal of Computer Vision, 14: pp.67-81, 1995.
- [9] B.Horn, B.Schunck: Determining optical flow, Artificial Intelligence, 17: pp.185-204, 1981.
- [10] K.Prazdny: On the Information in Optical Flows, Computer Vision, Graphics and Image Processing, 17, pp. 238-248, 1983.
- [11] J.Gibson: The Perception of the Visual World, Houghton Mifflin, 1950.
- [12] B.K.P.Horn, Robot Vision, MIT Press, Cambridge, MA, 1986.
- [13] D.Heeger: Optical Flow Using Spatiotemporal Filters, International Journal of Computer Vision, pp.279-302, 1988.
- [14] T.Tian, C.Tomasi, D.Heeger: Comparison of Approaches to Egomotion Computation, IEEE International Conference of Image Processing, pp.315-320, 1996.
- [15] B.Buxton, H.Buxton: Computation of optic flow from the motion of edge features in image sequences, Image and Vision Computing, 2: pp.59-75, 1984.
- [16] L.Telford, I.Howard, M.Ohmi: The effects of discordant visual-motor and visual-vestibular information on judgements of heading, Proceedings of European Conference on Visual Perception, 1992.
- [17] D.Lee, J.Lishman: Visual Proprioceptive Control of Stance, Journal of Human Movement Studies, 1: pp.87-95, 1975.