

インターレースカメラを利用した3次元空間での移動量推定手法

鈴木 雄貴[†] 塚本 昌彦[†] 國本 佳嗣^{††} 佐野 渉二^{††} 義久 智樹^{†††}

[†] 神戸大学大学院工学研究科

^{††} 神戸大学大学院自然科学研究科

^{†††} 京都大学学術情報メディアセンター

近年、カメラの小型化・高性能化に伴い、カメラから取得した画像をもとに解析を行い、移動量を推定する研究が盛んに行われている。多くの手法は、撮影された2枚の連続する画像を比較、解析することで移動量を推定するが、一般的な性能のPCにおいては、PC側の処理速度が追いついていないため、連続するフレーム間のスキャン間隔が長く高速な動きの推定が行えない。本研究では、マーカを利用せず、3次元空間を高速で移動する物体の移動量・移動方向の推定する手法を提案する。さらには、評価実験をすることにより有用性を確認する。

A Displacement Estimation Method by an Interlace Camera in 3-dimensional Space

Yuhki SUZUKI[†], Masahiko TSUKAMOTO[†], Yoshitsugu KUNIMOTO^{††},
Shoji SANNO^{††} and Tomoki YOSHIHISA^{†††}

[†] Graduate School of Engineering, Kobe University

^{††} Graduate School of Science and Technology, Kobe University

^{†††} Academic Center for Computing and Media Studies, Kyoto University

Recently, many researches are done on estimating the amount of movements of an object by using a fixed camera or an embedded camera. However, they do not support their high-speed movement because the scanning interval of the continuous frame is very long. In this paper, we propose a method of estimation of the amount of movements and the move direction which we do not use a marker and support their high-speed movement in three-dimensional space. We conduct the evaluation experiment and confirm the usefulness of our proposing method.

1 はじめに

近年、カメラの小型化・高性能化に伴い、カメラ画像から撮影対象やカメラの速度を推定し、それらの位置を把握する研究が盛んに行われている。例えば、ロボットに搭載されたカメラを用いて、ロボットの位置を推定し、災害救助や介護、イベント会場でのナビゲーションに役立つ研究 [1, 2, 3] が挙げられる。災害現場や介護現場などにおいては、平坦な所ばかりではないため、ロボットが障害物乗り越えたり、窪みに落ちて進んだりするなど、3次元空間での位置推定が必要となる。また、空中を飛行する小型ヘリ [3] などの小型ロボットに応用する場合も、3次元空間での位置の検出が不可欠である。

3次元空間での位置検出に関して、Stateら、Neumannら、大江ら、加藤らの研究がある [4, 5, 6, 7]。Stateらの研究 [4] では、2台のカメラと磁気センサーを併用した方法を提案している。2種類のセンサーを用いることでお互いの欠点を補っている。Neumannらは1台のカメラによる方法を提案している [5]。しかし、複数マーカへの検出が必要とされている。大江らの研究 [6] では、事前に撮影された動画画像から検出された自然特徴点をランドマークとして用いることでカメラ位置・姿勢推定手法を提案している。しかし、ランドマークと入力画像を対

応付けるために必要な情報として、撮影地点ごとの情報を保持する必要がある。加藤らは、拡張現実感システムのためのマーカ追跡手法とキャリブレーション手法を提案している [7]。以上の手法は、マーカやセンサを利用したり、撮影地点ごとの情報が必要であったりと、周りの環境に制限があり汎用性に欠ける。

筆者らの研究グループでは、インターレースカメラを利用した2次元空間の速度推定手法を提案している [8]。この手法では、2枚のカメラ画像からオプティカルフローを求めてカメラを搭載した物体、もしくは撮影対象物体の動きを把握しているため、センサやマーカを設置する必要がない。また、インターレーススキャンを利用することで、カメラが高速で移動する場合にも移動量の推定を行うことができる。本稿では、オプティカルフローベクトルについて、カメラに対して撮影方向、回転方向の移動も考慮することにより、カメラの3次元での移動に対応できるよう拡張し、インターレーススキャンを利用することでカメラの高速移動にも対応する手法を提案する。評価実験の結果、提案手法によりカメラが3次元空間の移動に対しても、その移動量を推定できることを確かめ、インターレーススキャンを利用することで高速移動に対応できることを明らかにした。

以降、2章でオプティカルフローを利用した動きの推定について述べ、3章で提案手法、4章で提案手法の評価実験の結果と評価を述べ、最後に5章で本稿をまとめる。

2 オプティカルフローを利用した動きの推定

2.1 オプティカルフロー

オプティカルフローは、画像中の各画素が次の瞬間にそれぞれどの方向にどれだけ動いたかを求めたベクトルである。

オプティカルフローベクトルの求め方にはいくつかの手法があるが、計算処理時間が短いとされているため、本稿では勾配法の中の手法の一つである Lucas-Kanade 法を用いた。

勾配法は、「連続する2枚の画像において移動物体の特徴を表す濃淡分布の移動量は微小である」という仮定を基にオプティカルフローを求める手法である。Lucas-Kanade 法は、1フレームを図1のような複数の局所領域に分け、連続するフレームの局所領域間の輝度の差や輝度の時間微分を求めることにより、移動方向、移動量を求める手法である。

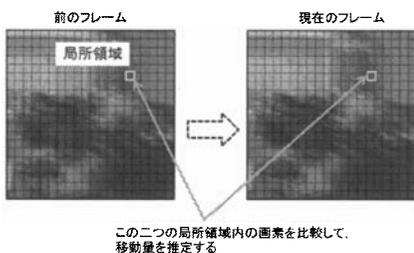


図 1: オプティカルフローを用いた推定手法

2.2 オプティカルフローを利用する場合の問題点

オプティカルフローを用いた移動量推定手法では、カメラを搭載した物体もしくは、撮影対象物体がある一定以上高速で移動する場合には、オプティカルフローベクトルを正確に求めることができない。その理由は主に二つあると考えられる。一つ目は、カメラを高速で移動させたときの撮影画像の歪みやぶれである。二つ目は、連続するフレーム間の局所領域内の画素に共通部分がないことである(図2)。図2(a)のように、連続するフレーム間の局所領域内の画素に共通部分がある場合は、移動量を推定することができる。しかし、図2(b)のように連続するフレームのある局所領域内の画素に共通部分がない場合は、オプティカルフローベクトルを

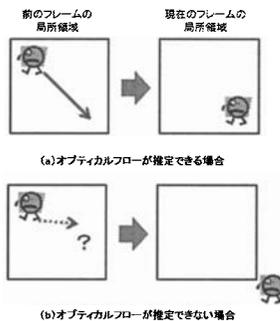


図 2: 従来のオプティカルフローの問題点

推定することができない。

一般的なカメラと一般的な PC を利用する場合、カメラのスキャン間隔に対して、PC の処理が追いついていないため、カメラでキャプチャした画像の中で数枚に1枚の画像しか PC に取り込むことができない(図3)。スキャン間隔とは、画像をスキャンし始めてから次の画像をスキャンし始めるまでの時間のことであり、ここでの PC の処理とは、カメラからの画像を PC に取り込む処理のことを指す。PC に取り込む連続する2枚の画像の、局所領域内の画素に共通部分なくなるほどカメラが高速で移動する場合は、正確に移動量を算出できない。例えば、一般的なカメラで VGA の画像をキャプチャする場合、カメラのフレームレートは 30fps 程度と考えられるので、連続する二つの画像間のスキャン間隔は約 33ms である。一方で、キャプチャした画像を PC 側に取り込むと、フレーム間のスキャン間隔が 100ms 程度になる。この PC に画像を取り込むスキャン間隔の長さが高速移動に対応できない要因となる。

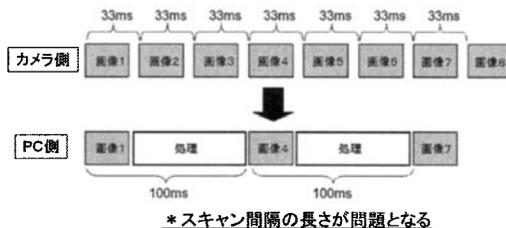


図 3: PC 側でのキャプチャの問題

3 提案手法

本研究では、3次元空間を高速で移動する対象物体の移動量・移動方向を推定する手法を提案する。

3.1 オプティカルフローを用いた移動量推定手法の3次元空間への拡張

物体の3次元空間での動きとは、カメラの撮影方向（上下方向）、その方向に対して垂直な平面方向（水平方向）の移動、物体自身の回転移動の三つがある。水平方向の移動に関しては、これまでに筆者らのグループが提案している手法 [8] のように、オプティカルフローベクトルをそのまま使用することで移動量を推定できる。以下に、回転、上下の動きの捉え方を示す。

1. 回転の推定手法

カメラを搭載した物体がカメラに対して右に回転する場合、図4のようなベクトルが得られる。得られた画像の中心点から横方向に引いた線の角度を0とする。また、図5のように、中心点とベクトルの始点・終点を線で結び、これらの線と角度0の線との角度をそれぞれ θ_1 、 θ_2 とし、それらの差を $\theta = \theta_2 - \theta_1$ とすると、 $\theta < 0$ となり、 θ の大きさにより移動量が推定できる。カメラを搭載した物体が左回転する場合は、逆に $\theta > 0$ となる。 θ の正負の符号、絶対値の大きさによりカメラを搭載した物体の回転方向・移動量を推定する。

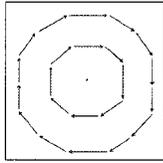


図4: カメラを搭載した物体が右回転する際のベクトル

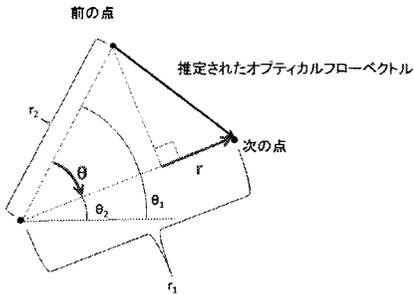


図5: カメラが回転・上下移動するときの移動量推定

2. 上下移動の推定手法

カメラを搭載した物体がカメラの撮影方向に移動する場合、カメラに映っている対象は大きくなるため、図6のようなオプティカルフローベクトルが得られる。このとき、図5のように、得られた画像の中心点とベクトルの始点・終点との距離をそれぞれ r_1 、 r_2 とし、それらの距離の差を $r = r_2 - r_1$ とすると、 $r > 0$ となり、 r の大きさにより移動量が推定できる。カメラの撮影方向と逆方向に移動する場合は、逆に $r < 0$ となる。 r の正負の符号、絶対値の大きさによりカメラを搭載した物体の上下の移動方向・移動量を推定する。

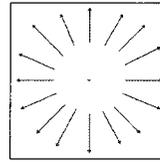


図6: カメラの撮影方向に移動する際のオプティカルフローベクトル

3.2 インターレースカメラを用いた移動量推定手法

3.2.1 インターレーススキャン

カメラが一つの画像をスキャンするときに、水平方向の走査線を上から下に走査する。インターレーススキャンとは、一つの画像（フレーム）の表示を1回目に奇数走査行、2回目に偶数走査行という2回に分けてスキャンする方式のことである。また、一つの画像を1回の走査でスキャンする方式をノンインターレース方式という。例えば、ノンインターレースカメラのフレームレートが30fpsである場合、インターレーススキャンの奇数走査画像と偶数走査画像を別画像とみると、インターレースカメラのフレームレートは60fpsに相当する。

3.2.2 高速移動に対応するための手法

カメラでキャプチャしたVGAの画像を一般的な性能のPCに取り込む場合、連続する2フレームの画像のスキャン間隔は約100msと長い。このスキャン間隔の長さを克服するため、提案手法ではカメラのインターレーススキャンを利用する。インターレーススキャンの奇数走査、偶数走査間のスキャン間隔は、連続する2フレーム間の画像のスキャン間隔に比べると非常に短い。上記の100msに1枚しか画像が得られない場合でも、インターレーススキャンを利用することで、1枚の画像から更新周期の非常に短い2枚の画像を得ることが可能となる。2枚の画像を利用しオプティカルフローベクトルを

求めることで、高速で移動する物体の動きを捉えることができる (図 7)。

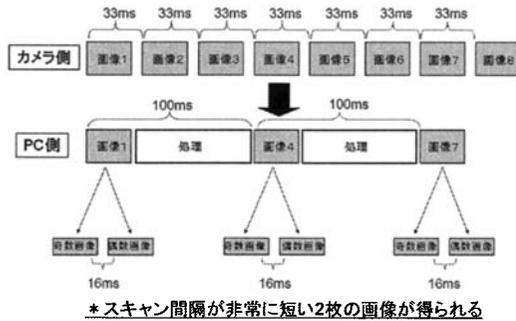


図 7: 高速移動に対応する方法

3.3 提案手法の移動量推定の流れ

提案手法の流れを以下に示す (図 8)。カメラの解像度は 480×680 の RGB カラー画像とする。

1. 画像をキャプチャする。
インターレースカメラにより画像を取得する。
2. キャプチャした画像の解像度を変更する。
オプティカルフローを求める際に使用する画像の大きさは、縦横同じ画素数である方が精度の誤差が少ないと考え、4. で 2 分割させることを考慮し、 480×240 の大きさの画像に直す。このとき、最近隣内挿法 (Nearest Neighbor Method) を用いてリサンプリング処理を行う。最近隣内挿法とは、最も近いピクセルの値を利用して出力ピクセルに割り当てる手法である [9]。
3. 画像をグレースケール画像に変換する。
処理を単純にして処理速度を上げるため、カラー画像をグレースケール画像に変換する。カメラが取得する画像の輝度は $0 \sim 255$ の値をとることになる。
4. 奇数走査・偶数走査画像に分ける。
グレースケール画像を、図 8(b) のように奇数走査により得られた画像と偶数走査により得られた画像に分ける。画像の縦方向に関し、1 行飛ばして画像をスキャンしているため、ここで得られる画像サイズは 240×240 である。
5. それら 2 枚の画像からオプティカルフローを推定する。
4. で得られた 2 枚の画像から、Lucas-Kanade 法を用いて偶数走査画像から奇数走査画像へのオプティカルフローを推定する。

6. 推定したオプティカルフローベクトルを画像として保存する。
オプティカルフローベクトルを保存する画像サイズは 240×240 となる。

7. 各点でのオプティカルフローベクトルを描画し、各フレームでのフローベクトルを求める。
保存されたフローベクトルの画像を、キャプチャした画像上に描画する。5. で、キャプチャされた 1 枚の画像の各点におけるフローベクトルが求められた。それらを 1 フレーム内で合計し、そのフレームでのオプティカルフローベクトルとする (図 8 (c))。

保存されたフローベクトルの画像を、キャプチャした画像上に描画する。5. で、キャプチャされた 1 枚の画像の各点におけるフローベクトルが求められた。それらを 1 フレーム内で合計し、そのフレームでのオプティカルフローベクトルとする (図 8 (c))。

8. 得られたフローベクトルを用いて、3 次元空間での物体の移動量・移動方向を推定する。
カメラを搭載した物体もしくは撮影対象の 3 次元空間での移動量・移動方向を推定できる。

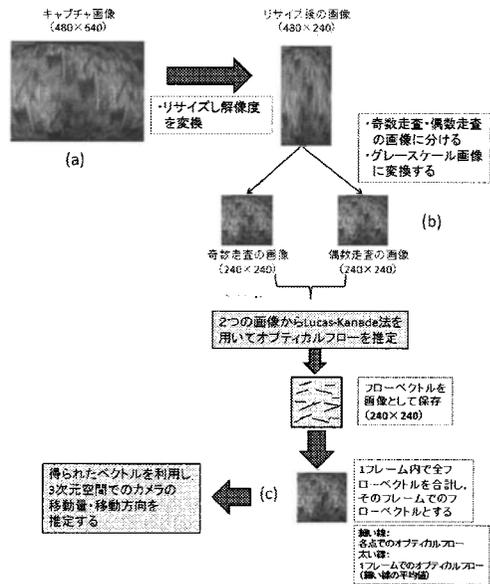


図 8: 提案手法のオプティカルフローベクトルの推定方法

4 評価実験

4.1 実験環境

1. カメラは一般的な性能のものを利用する

カメラの性能を上げることで、フレーム間のスキャン間隔は短くなるため、高速移動へ対応することができる。しかし、様々な分野へ

応用する際、必ずしも高性能なカメラを使えるとは限らない。本研究では、汎用性を持たせるため一般的な性能のカメラを利用する。実験に使用したカメラは、アールエフ株式会社製の「The CARD-7RL」である（図9）。このカメラはインターレース方式で画像のスクランを行い、奇数走査画像、偶数走査画像を別々の画像と見なすと、フレームレートは60fps、解像度は480×680である。



図9: 使用したインターレースカメラ

2. PCは一般的な性能のものを利用する

PCに取り込まれるフレーム間のスクラン間隔は、PCの処理速度が速い高性能なPCを用いることで短縮することができる。しかし、高性能なPCはサイズが大きいため、人やロボット、小型ヘリなど移動する対象物体に取り付けるのは現実的ではない。本研究では、汎用性を持たせるため一般的な性能のPCを利用する。使用したPCのスペックを図10に示す。

CPU	Athlon(tm) 64 Processor 3500+ 986 MHz
メモリ	960 MB RAM

図10: 使用したPCのスペック

4.2 実験装置・実験手順

4.2.1 水平方向の評価実験

実験装置の様子を図11に、模式図を図13に示す。背景画像が添付された、手で回転させることができる円柱状の筒（直径40.5cm（円周約127.17cm））を使用する。カメラで背景画像全体を十分に撮影できるように、筒から上33cmの距離にカメラを固定し、筒を動かすことでデータを取る。円筒状の筒には、図14に示すようなカラー画像を添付する。この画像は画像処理でよく用いられる標準画像であり、インテル社が公開している画像処理ライブラリOpenCvに含まれているものを、JPEGで348KB（2446ピクセル×2446ピクセル（64.71cm



図11: 実験装置



図12: カメラのx座標、y座標

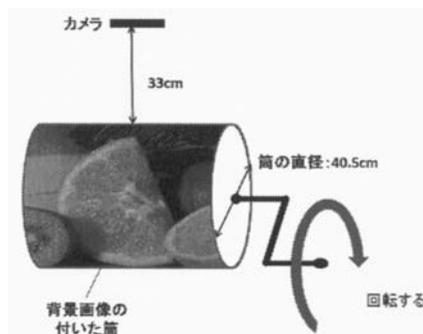


図13: 水平方向の実験装置模式図



図14: 実験で使用した画像

×64.71cm))に圧縮した。この画像を、プリンタ「canon MP 900」により、A4サイズの上質紙に分割して印刷し、張り合わせて用いた。この環境において、実際の10cmをカメラで撮影すると95ピクセルに相当する。

背景画像が添付された筒を一定方向に回し、推定されたオプティカルフローベクトル、フレーム間のスクラン間隔のデータを取る。カメラに対してyの正方向へ動かし、yの正方向がオプティカルフローの出る方向とした（図12）。

回転速度を変えながら同じ操作を90回行い、従来手法・提案手法ともに、筒を動かす速度とオプティカルフローの大きさ（検出値）をグラフにプロットする。また、スクラン間隔からフレームレートを割

り出す。

4.2.2 回転方向の評価実験

図 15 に示すように、カメラを回転する器具に取り付け、カメラで背景画像全体を十分撮影できるように、カメラから 40cm のところに図のような背景画像を立てる。このときカメラの中心と背景画像の中心を一致させる。背景画像には、図 11 の画像を用い、JPEG で 296KB に圧縮した (3081 ピクセル×3081 ピクセル (81.51cm × 81.51cm))。この環境において、実際の 10cm をカメラで撮影すると 50 ピクセルに相当する。

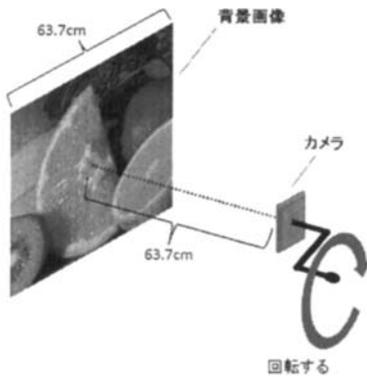


図 15: 回転方向の実験装置模式図

カメラを図 15 で示す方向に回転させ、推定されたオプティカルフローベクトル、フレーム間のスキャン間隔のデータを取る。回転速度を変えながら同じ操作を 120 回行い、従来手法・提案手法ともに、筒を動かす速度とオプティカルフローの大きさ (検出値) をグラフにプロットする。また、スキャン間隔からフレームレートを割り出す。

4.2.3 上下方向の評価実験

図 16 に示すように、台車の高さ 70cm のところに地面と垂直にカメラを取り付け、カメラで背景画像全体を十分撮影できるように、カメラから 90cm 離れた所に図のような背景画像を立てる。このとき、カメラの中心と背景画像の中心を一致させる。背景画像には、図 14 の画像を用い、JPEG で 296KB に圧縮した (3081 ピクセル×3081 ピクセル (81.51cm × 81.51cm))。この環境において、実際の 10cm をカメラで撮影するとカメラが背景画像から最も遠いとき 30 ピクセル、最も近いとき 240 ピクセルに相当する。

カメラを搭載した台車を、背景から 90cm 離れた場所から 10cm のところまで近づけ、推定されたオプティカルフローベクトル、フレーム間のスキャン

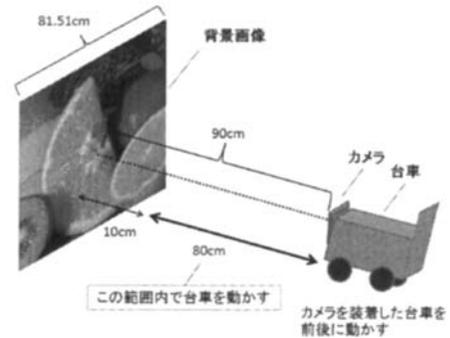


図 16: 上下方向の実験装置模式図

間隔のデータを取る。台車の速度を変えながら同じ操作を 90 回行い、従来手法・提案手法ともに、筒を動かす速度とオプティカルフローの大きさ (検出値) をグラフにプロットする。また、スキャン間隔からフレームレートを割り出す。

4.3 従来手法のアルゴリズム

提案手法で利用しているインターレーススキャンを評価するために、ノンインターレースカメラを利用した際に得られる、2 フレームの画像を用いてオプティカルフローベクトルを推定する手法に近い形とする。本研究で用いているカメラはインターレースカメラであり、奇数走査画像と偶数走査画像に分けるため、片方の画像のみを用いれば 2 フレームの画像に相当するものとみなせる。本評価ではインターレーススキャンを用いない方法として、奇数走査画像だけを利用することで比較する。PC に取り込む画像のフレームレートを等しくするため、提案手法とインターレーススキャンを用いない手法は、同じプログラム内で実行し、同時にデータを取る。

4.4 実験データの取り方

連続する 2 フレームにつき得られるオプティカルフローベクトルの値は一つである。しかし、1 回の実験においては複数フレームの画像がカメラでキャプチャされるため、ベクトルの値も複数個得られることになる。フレーム 1 からフレーム n までのベクトル値が得られた場合、それらの平均値を 1 回の実験データとして扱う。また、フレーム間のスキャン間隔も n 個得られるため、それらの平均値を 1 回の実験データとする。

4.5 実験結果・考察

カメラもしくは撮影対象物体の移動速度が速いほど、推定されるオプティカルフローの値は大きくなるため、速度とオプティカルフローの検出値が比例

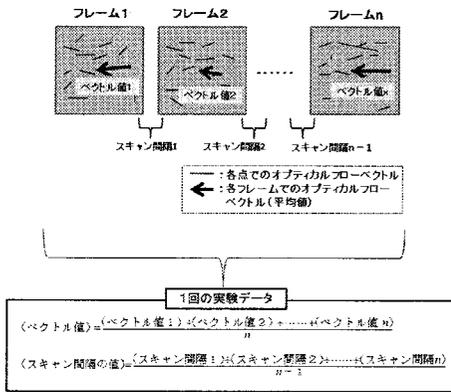


図 17: 実験データの取り方

関係にあるとき、正確な移動量が求められていると判断できる。

4.5.1 水平方向の評価

実験データを図 18 に示す。グラフの横軸が背景を添付した筒の移動速度、縦軸がオプティカルフローのベクトルの大きさを示している。PC に画像を取り込むフレームレートは 5.87fps、照度は 1000LUX である。

円周 127.17cm の筒に 1 種類の背景画像が 2 枚添付されているため、カメラが 2 枚の画像をキャプチャする間に、筒が半周 (約 63.69cm) 以上回ると、2 枚の撮影画像が全く同じになってしまう、測定誤差につながる恐れがある。実験結果より、連続する 2 フレーム間のスキャン間隔の平均値は約 170.23[ms] であり、筒の移動速度が約 374.14[cm/s] であるときにその現象が起こるため、実験データからこの測定誤差は発生していないと考えられる。

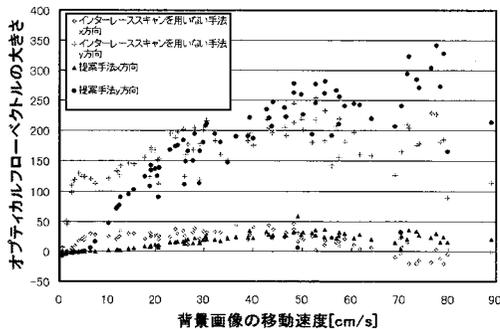


図 18: 水平方向の実験結果

背景を添付した筒は一定方向 (y の正方向) へ動かしているため、誤差がまったくないと考えた場合、オプティカルフローの検出値は y の正方向の値のみ出ることになる。もし、 x 方向への値または y の負方向への値が出ている場合は、オプティカルフロー推定に誤差が生じており、正確な速度推定が行えていないと考えられる。

図 18 より、オプティカルフローベクトルが出るべき方向である y のデータを見ることにより、提案手法の限界速度 (正確な移動量推定が行えなくなる速度) を求めると、約 55.0[cm/s] であり、水平方向の移動量推定が行えていることが分かる。一方、インターレーススキャンを用いない手法の限界速度は、約 25.0[cm/s] である。提案手法は、これの約 2.2 倍であるため、インターレーススキャンを用いることで高速移動に対応できていることが示された。理想では出るべきでない x 方向のデータを見ると、どちらの手法ともに、 y 方向と比較すると小さい値ではあるが検出されている。これは、実験装置による誤差であると考えられる。

4.5.2 回転方向の評価

実験データを図 19 に示す。グラフの横軸がカメラの回転速度、縦軸がオプティカルフローのベクトルの大きさを示している。PC に画像を取り込むフレームレートは 5.70fps、照度は 1000LUX である。

カメラが 2 枚の画像をキャプチャする間に、カメラ自身が 1 回転以上回転すると、2 枚の撮影画像が全く同じになってしまう、測定誤差につながる恐れがある。実験結果より、連続する 2 フレーム間のスキャン間隔の平均値は約 175.40ms であり、カメラの回転速度が約 2052.45[°/s] であるときにその現象が起こるため、実験データからこの測定誤差は発生していないと考えられる。

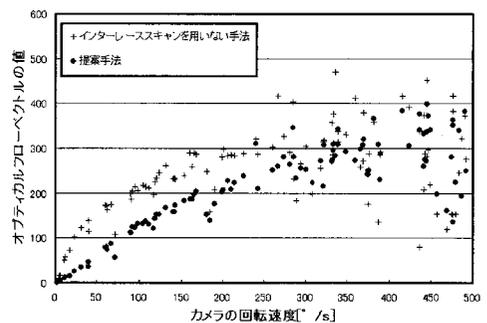


図 19: 回転方向の実験結果

図 19 より、提案手法の限界速度を求めると提案手法は約 370.0[°/s] であり、回転方向の移動量推定が行えていることが分かる。一方、インターレースス

キャンを用いない手法の限界速度は、約 $170.0[^\circ/s]$ である。提案手法は、これの約 2.18 倍であるため、インターレーススキャンを用いることで高速移動に対応できていることが示された。尚、データがばらついている部分があるが、これは実験装置による誤差であると考えられる。

4.5.3 上下方向の評価

実験データを図 20 に示す。グラフの横軸がカメラの回転速度、縦軸がオブティカルフローのベクトルの大きさを示している。PC に画像を取り込むフレームレートは 5.78fps、照度は 1000LUX である。

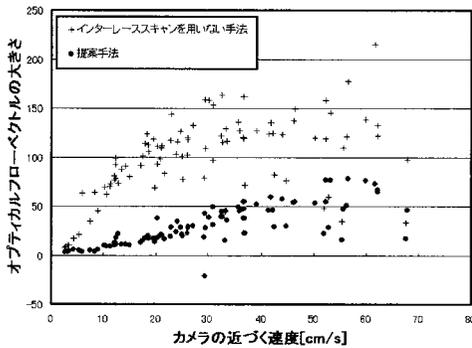


図 20: 上下方向の実験結果

図 20 より、提案手法の限界速度を求めると提案手法は約 $2.0[^\circ/s]$ であり、上下方向の移動量推定が行えていることが分かる。一方、インターレーススキャンを用いない手法の限界速度は、約 $4.0[^\circ/s]$ である。提案手法は、これの約 2.00 倍であるため、インターレーススキャンを用いることで高速移動に対応できていることが示された。尚、データがばらついている部分があるが、これは実験装置による誤差であると考えられる。

5 おわりに

本研究では、3次元空間においてカメラで高速で移動する物体の移動量を推定するために、インターレースカメラとオブティカルフローを用いた移動量推定手法を提案した。1フレーム内の奇数走査画像と偶数走査画像を2枚の画像と見なし、得られたオブティカルフローベクトルを極座標変換することにより、3次元空間での物体の移動量を推定する。さらに、インターレーススキャンを用いることで、高速移動にも対応できる。評価実験を行うことで、提案手法の有用性を確認した。今後の課題としては、カメラ自身もしくは撮影対象の速度や位置推定を行

い、カメラを搭載したラジコンヘリコプタの自動制御、カメラを用いた空中マウスなどへ応用する。

謝辞 本研究の一部は、連携創造本部先端研究推進部門の学内共同研究によるものである。ここに記して謝意を表す。

参考文献

- [1] G. Gartner: Location-based mobile pedestrian navigation services - the role of multimedia cartography, ICA UPIMap, pp. 7-9, 2004.
- [2] 畑瀬 勉, 若松 善美, 西村 英記, 山本 博利: インテリジェント車いすロボット, FUJITSU, Vol. 57, No. 3, pp. 263-268, 2005.
- [3] 塚本 昌彦, 松坂 敬太郎: 飛ぶコンピュータの実現に向けて, 情報処理学会第 39 回モバイルコンピューティングとユビキタス通信 (MBL) 研究会報告, No. 120, pp. 111-118, 2006.
- [4] A. State, G. Hirota, D. T. Chen, W. F. Garrett, M. A. Livingston: Superior Augmented Reality Registration by Integrating Landmark Tracking and magnetic Tracking, Proceedings of SIGGRAPH96, pp. 429-446, 1996.
- [5] U. Neumann, S. You, Y. Cho, J. Lee, J. Park,: Augmented Reality Tracking in Natural Environments, Ohmsha and Springer-Verlag, pp. 101-130, 1999.
- [6] 大江 純子, 佐藤 智和, 横矢 直和: 画像特徴点によるランドマークデータベースに基づくカメラ位置・姿勢推定, MIRU2005 講演論文集, pp. 800-807, 2005.
- [7] 加藤 博一, M. Billingham, 浅野 浩一, 橘 啓八郎: マーカ追跡に基づく拡張現実感システムとそのキャリブレーション, 日本バーチャルリアリティ学会論文誌, Vol. 4, No. 4, pp. 607-616, 1999.
- [8] 鈴木 雄貴, 國本 佳嗣, 佐野 涉二, 義久 智樹, 塚本 昌彦: インターレースカメラによるフレーム内オブティカルフローを用いた高速移動対応の速度推定手法, 情報処理学会マルチメディア, 分散, 協調とモバイルシンポジウム (DICOMO 2007) 論文集, Vol. 2007, No. 1, pp. 293-302, 2007.
- [9] 村上伸一: 理工学講座: 画像処理工学, 東京電機大学出版局, 1996.