論文

球技における最適動作推定のための

画像処理による空気抵抗測定

Air Resistance Measurement by Image Processing for Optimum Motion Estimation in Ball Games

**概要**

　実際のバドミントンの練習動画から，シャトルの軌道を取得するプログラムを開発した．シャトルの物理モデルを考え，速度ベルレ法を用いた数値シミュレーションを行い，シャトルの空気抵抗係数を求めた．

**Abstract**

**1.はじめに**

**1.1.本研究の背景**

　球を取り扱い，技術を競う球技スポーツにおいて，球の特性は最も重要な要素であり，特性を知ることはスポーツ技術の向上に繋がる．また，球の打ち方やそのフォームも重要な要素であり，目的に応じて最適な動作を行うことが求められる．ただ，その動作に関して, コーチの経験的な指導のみでは不十分という問題がある．そこで，球を打つときの角度などの客観的な指標を力学的な視点から解析することが必要とされている．その解析手法の１つとして，人や球の物理モデルを構築し，数値計算を行うコンピューターシミュレーションが挙げられる．その際，慣性モーメントや空気抵抗係数などの物理モデルの特性を決定するパラメータが非常に重要となる．

　本研究では，全ての球技の中で初速度が最も大きいバドミントンのシャトルに着目し，その特性を調べる．現在，シャトル軌道推定や，空力特性に関する研究の報告はあるが，シャトルの力学的なモデルを考え，運動方程式を解くことによりパラメータを同定している研究は少ない．物理モデルを構築することにより，シャトルの運動が記述出来れば，シャトル軌道推定やライン判定などに生かせるため，スマッシュの最適な打ち出し角や，ライン際を狙ったショットの効果的なトレーニングが期待できる．

**1.2.本研究の目的**

　本研究では, 球技における最適動作推定のために，球の特性を求めることを目的としている．対象は，球技の中でも形状が比較的複雑で，激しい減速特性を持つバドミントンのシャトルとする．シャトルの物理モデルを考案し, シミュレーションを用いて空気抵抗係数などのシャトルの特性を表すパラメータを求める. パラメータは，バドミントンの練習動画から取得した現実のシャトル軌道と，シミュレーションでの軌道を照らし合わせながら調整し，現実のシャトル軌道を描けるようなパラメータを導き出す．

**1.3.本論文の構成**

　本論文の構成を以下に示す．基本技術と原理では，モデル化，速度ベルレ法，シャトルの軌道検出方法について述べる．評価実験では，シミュレーションによるパラメータの調整方法について述べる．考察では，パラメータの調整結果を評価し，考察する．最後に，結論では本研究の結論を述べるとともに，今後の課題について検討する．

**2.基本技術と原理**

**2.1.シャトルのモデル化**

　シミュレーションに用いるシャトルは, 重力と空気抵抗の2つの力が加わるモデルとした．図1にモデルを示す．形状による空気抵抗の異方性を考慮するため，シャトルの水平方向と鉛直方向の2方向で異なる空気抵抗を加えた．

　また，シャトルに回転力を与えるために重心と抵抗力の作用点を だけずらしている．それに伴って，シミュレーションでは並進と回転の運動方程式を同時に解く．簡単のため，シミュレーションの際はシャトルを質点に近似する．

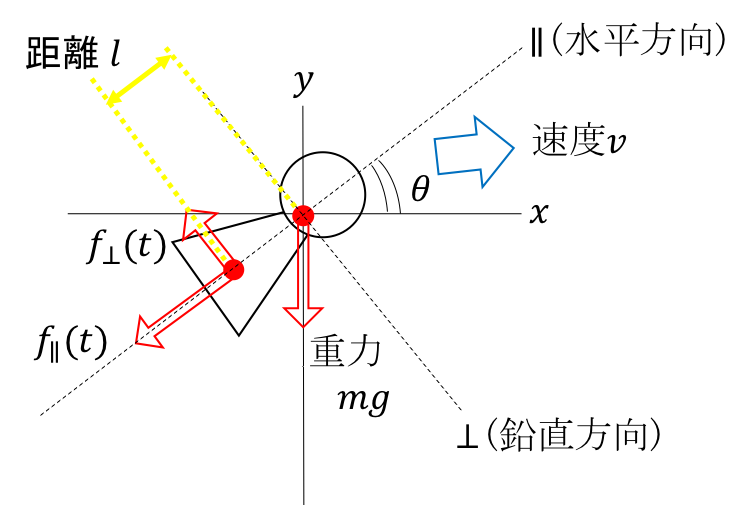


図1 シャトルのモデル

Fig.1 Shuttle’s model

　空気による抵抗力は，速度に比例する粘性抵抗と，速度の2乗に比例する慣性抵抗の2種類を考える．シャトルの羽根による減速特性を加味して，このモデルでは，鉛直方向に粘性抵抗，水平方向に粘性抵抗と慣性抵抗の両方を加えることとする．

　以下にそれぞれの抵抗力の式を示す．

鉛直方向：

　　　　　　　　 水平方向：

　ここで, ：粘性項の空気抵抗係数，：慣性項の空気抵抗係数，：重力項である．

**2.2.時間進展アルゴリズム**

　速度ベルレ法 (Velocity-Verlet algorithm) を用いて，シャトルの運動方程式を数値的に解くことにより，シミュレーションを行う．分子動力学法の数値積分手法として，広く使われている手法である．図2に今回用いた速度ベルレ法のフローチャートを示す．

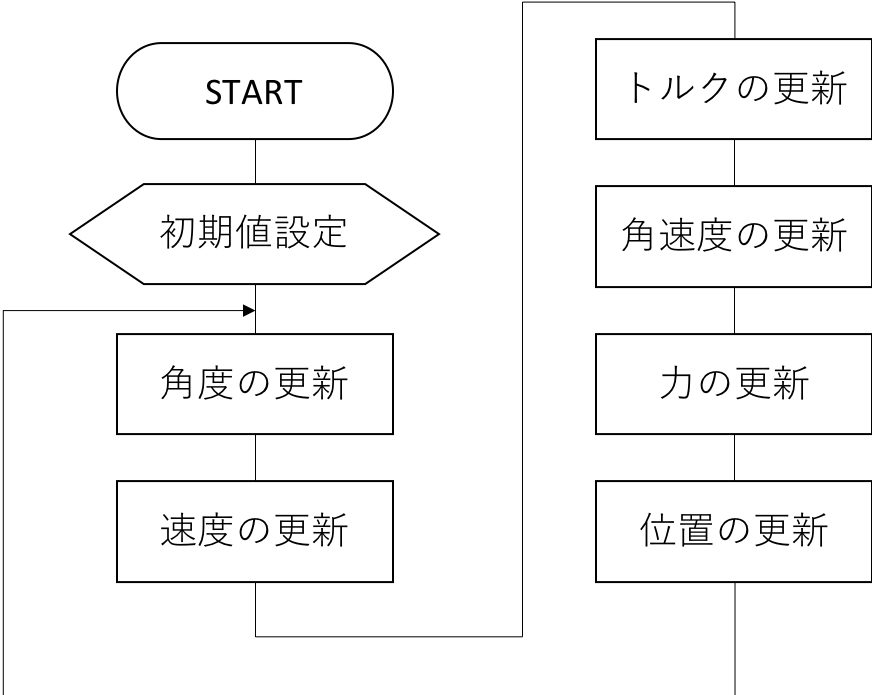


図2 速度ベルレ法

Fig.2 Velocity-Verlet algorithm

更新式の導出を以下に示す．

まず，位置 を 近傍でテイラー展開すると，

となり，物理量との関係から2次の項までを取り扱うと，

となる．第2項は速度項であり，第3項は加速度項である．

ここで，，運動方程式より であるから，

となる．この(3)式により，位置の更新を行う．

また， についても同様にテイラー展開すると

となり， と時間進展すると，

となる．この式に(3)式を代入し，左辺に を移項すると，

が導かれる．本来はこの(4)式により，速度の更新を行う．しかし，今回は慣性抵抗を考えるため，力 に速度の2乗の項が含まれ， の形への式変形が困難であった．よって，速度の更新式はテイラー展開の1次の項までを取り扱うことで導く．

速度 を 近傍でテイラー展開し，1次の項まで取り扱うと，

ここで，運動方程式より であるから，

この(5)式により速度の更新を行う．

位置や速度は，成分に分解して，それぞれ更新を行う．ただ，今回のモデルではシャトルから見て鉛直方向と水平方向の抵抗力を加えるので，速度更新に関しては，一旦鉛直，水平座標系に直す必要がある．図3に座標系の変換について示す．

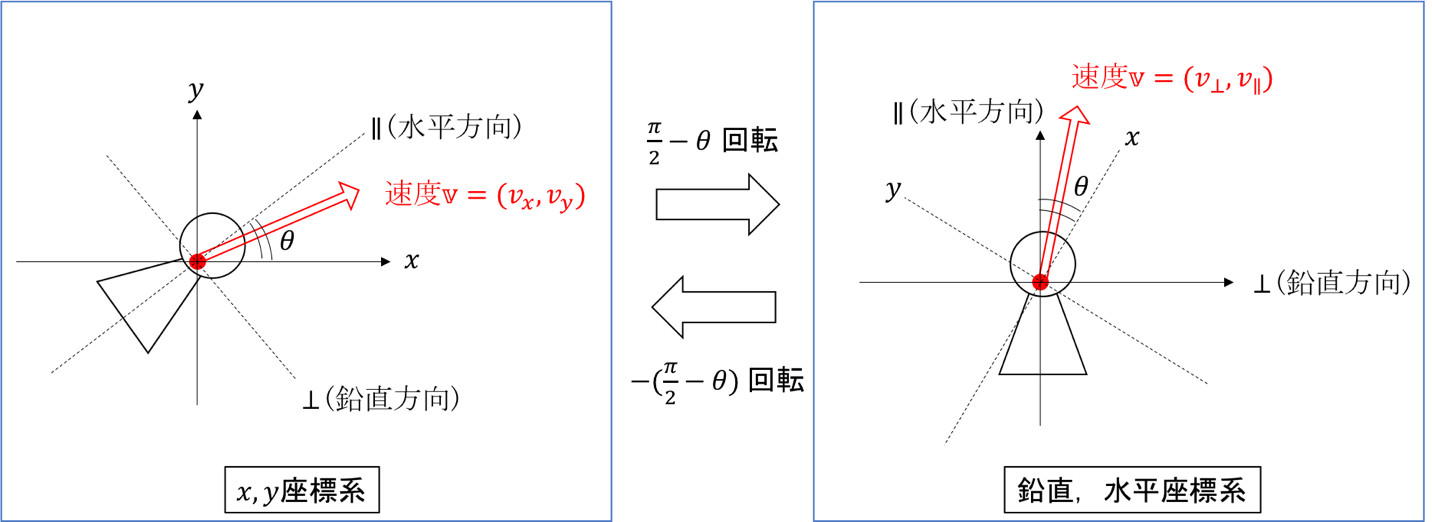


図3 座標系の変換

Fig.3 Transformation of coordinate system

速度 を，回転行列 を用いて だけ回転させることにより，に変換する．そして，(1)，(2)式で力を計算し，(5)式で に更新する．再度，回転行列 を用いて，だけ回転させることで座標系に戻す．

以上の手順で，速度 を へと更新する．

一方，並進の運動方程式 と回転の運動方程式 を比較すると，質量と慣性モーメント，位置と角度，力とトルクの対応が取れるので，(3)，(4)式より，

このように式が導ける．この(6)，(7)式により角度，角速度を更新する．

また，図1に示した，空気抵抗力の作用点と重心の距離 を用いて

この(8)式でトルクの計算を行う．

**2.3.シャトルの軌道検出方法**

　シャトル軌道検出のフローチャートを図4に示す．

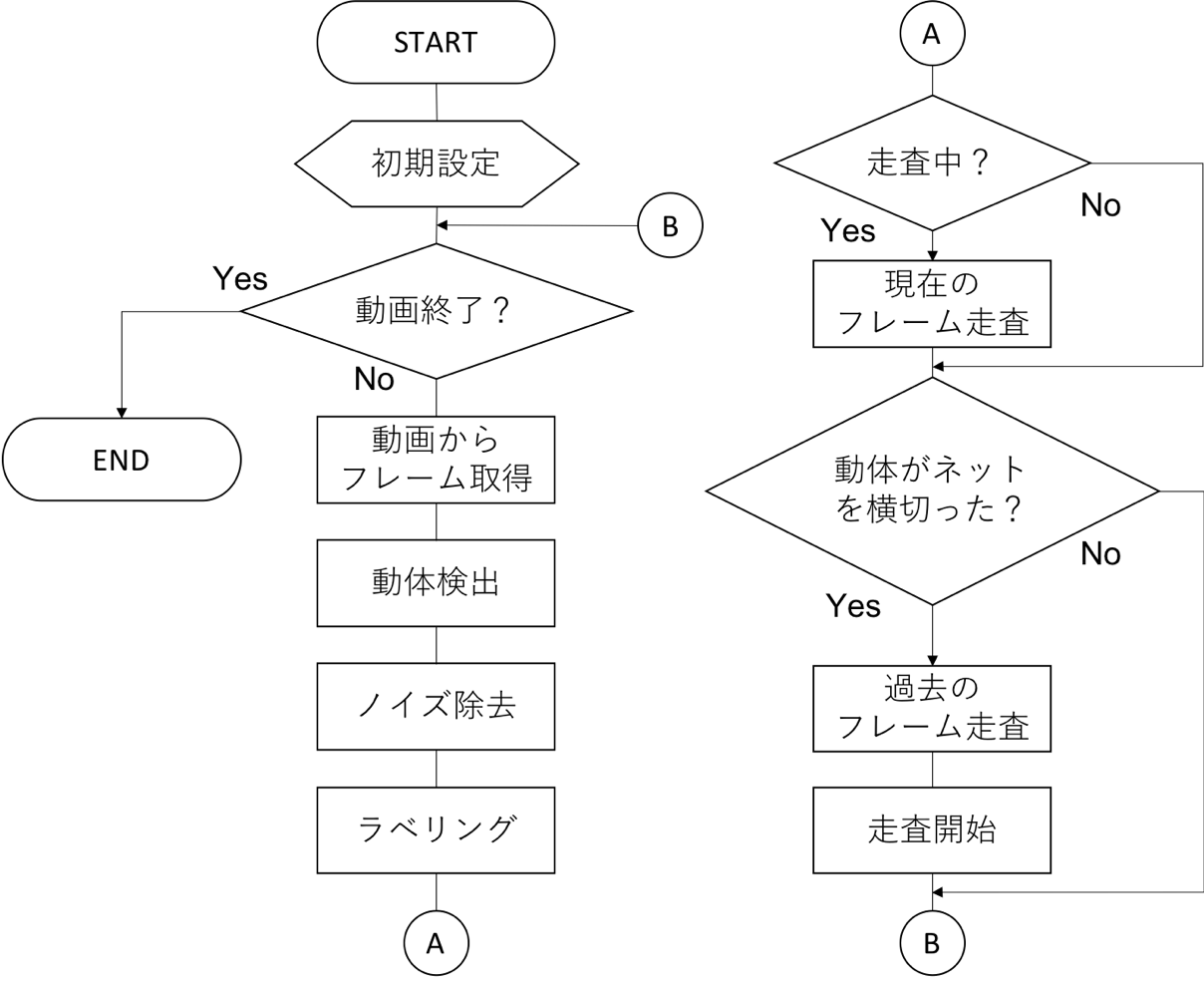


図4 シャトル軌道検出

Fig.4 Shuttle trajectory detection

　プログラム言語はC++，画像処理に用いたライブラリはOpenCV3である．動体検出は，動画から連続した3フレームを取得し，その差分を取ることで行う．また，erode関数とdilate関数により縮小と膨張を1回ずつ行うことでノイズを除去する．ラベリングは，connectedComponentsWithStatus関数を用いて，領域の面積と重心座標も取得する．

　動画のフレーム，動体検出，ノイズ除去，ラベリングの様子を表す画像を以下図5，図6，図7，図8に示す．ラベリングでは領域の区別を分かりやすくするため，領域ごとに色分けしている．



図5 フレーム画像

Fig.5 Frame image



図6 動体検出

Fig.6 Moving object detection



図7 ノイズ除去

Fig.7 Noise removal



図8 ラベリング

Fig.8 Labeling

シャトル検出のイメージを図9に示す．

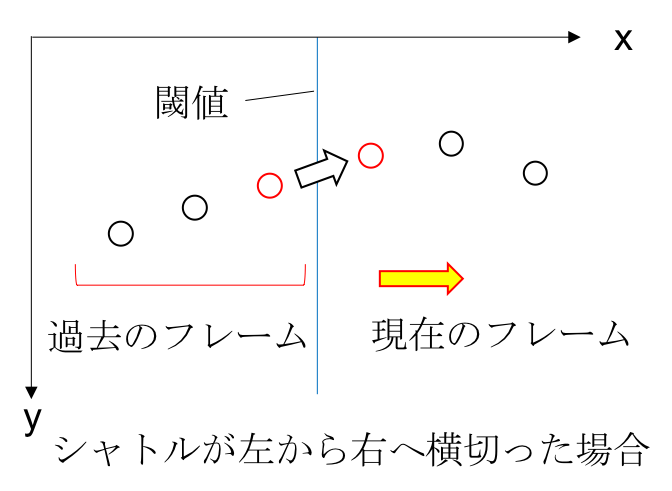


図9 シャトル検出

Fig.9 Shuttle detection

ネットの座標を閾値として，それをまたぐ動体をシャトルとして検出する．過去のフレームは保存しておき，シャトル検出時に，フレームを遡って軌道を取得する．フレームを遡る際，座標が近いか，と領域の面積が似ているか，の2点を調べて走査していく．この2点を満たす動体が無くなった時点で走査を打ち切る．過去フレームからの，順にシャトル近傍を追っていき，軌道とする．例えば，シャトルが左から右へ横切った場合，左側の軌道をシャトル検出時に取得する．

**3.評価実験**

**3.1.実験条件**

　計算機