

片側ラドン変換を用いたビデオフレームレートでの 物体認識と平面運動計測

Video-frame Rate Object Identification and Planar Motion Detection using One-sided Radon Transform

○坪井 辰彦 平井慎一 (立命館大学)

Tatsuhiko Tsuboi and Shinichi Hirai

Ritsumeikan University, Noji-higashi 1-1-1, Kusatsu, Shiga 525-8577

Abstract: In this paper, we will propose a vision algorithm using one-sided Radon transform.

First, we will introduce one-sided Radon transform. Second, we will develop a vision algorithm to detect planar motion of a rigid body. Finally, the developed vision algorithm is evaluated experimentally.

Key Words: vision, sensing, motion, identification

1. はじめに

本研究では、片側ラドン変換を用いたビジョンアルゴリズムを使用してビデオフレームレートでの物体認識と平面運動計測を行う。物体をハンドリングする際、視覚情報を用いたハンドリング制御が有効であると考えられる。物体ハンドリングにおいては、特に平面運動の検出が重要である。そこで、視覚情報を用いた平面運動物体を認識するアルゴリズムを構築する。

2. 平面運動検出ビジョンアルゴリズム

本報告では、平面運動する物体の種類を識別し、運動を計測するビジョンアルゴリズムを構築する。このアルゴリズムの用途として、物体の仕分けや箱詰めが挙げられる。

本報告で構築するビジョンアルゴリズムは、NTSC 方式のビデオ信号を入力として使用するため、物体の種類、位置・姿勢の計算を 33 [msec]以内で行う必要がある。

2.1 従来のアルゴリズム

従来の平面運動検出アルゴリズムは、

- ① 正規化相関演算法[1]
- ② テンプレート画像から回転画像を生成し正規化相関演算を適用する方式
- ③ 輪郭法[2]
- ④ ハフおよびフーリエ変換を用いた方法[3]

が挙げられる。まず、①は、物体の平行移動を認識できる。しかし、物体の回転を認識できない。②は、回転と平行移動を認識することができる。しかし、演算量は膨大になる。③は、平行移動、回転を検出することができ、演算量は①②と比較して少ない。しかし、対象物体は凸形状に限定され、凹形状や穴のある形状には適用できない。④は、拡大・回転平行移動を検出することができ、また、対象物体の形状に制限はない。しかし、ハフ変換とフーリエ変換を用いるため、演算量が極めて多い。

2.2 ラドン変換法

従来より、ラドン変換を用いて、物体の位置・姿勢を検出する手法が知られている。Fig1-(a)に示すように連続画像 $g(x,y)$ に対するラドン変換 $R_0(\theta, \rho)$ は、

$$R_0(\theta, \rho) = \int_{-\infty}^{\infty} g(\xi \cos \theta - \rho \sin \theta, \xi \sin \theta + \rho \cos \theta) d\xi \quad (1)$$

で与えられる。ラドン変換 $R_0(\theta, \rho)$ は、原点からの距離 ρ 、 x 軸からの角度が θ の直線における画素値の線積分である。入力画像のラドン変換とパターン画像のラドン変換を 2 次元パターンマッチングすることにより、物体の位置と姿勢を計算することができる。しかし、2 次元パターンマッチングは、多くの演算量を要し、リアルタイム処理には向かない。そこで、物体の位置を 2 値化重心計算により求め、物体の姿勢のみをラドン変換から求める手法が提案されている。この手法では、 $\rho = 0$ に対応するラドン変換を比較して、物体の姿勢を計算する。ただし、ラドン変換においては、Fig.2-(a)に示すように $R_0(\theta, 0) \equiv R_0(\theta + \pi, 0)$ が成り立つため、角度 θ と $\theta + \pi$ を区別することができない。

2.3 片側ラドン変換

ラドン変換においては、積分路が Fig.1-(a)に示すように直線であるため、角度 θ と $\theta + \pi$ を区別することが困難である。そこで、積分路を直線から、Fig.1-(b)に示すように半直線に変更する。すなわち、

$$R(\theta, \rho) = \int_0^{\infty} g(\xi \cos \theta - \rho \sin \theta, \xi \sin \theta + \rho \cos \theta) d\xi \quad (2)$$

とする。これを片側ラドン変換と呼ぶ。連続画像 $g_1(x,y)$ を原点周りに角度 α 回転させた画像を $g_2(x,y)$ とする。また、 $g_1(x,y)$ を方向 β 、距離 d_0 並進移動させた画像を $g_3(x,y)$ とする。このとき、 g_1, g_2, g_3 の片側ラドン変換をそれぞれ R_1, R_2, R_3 で表すと、次式が成り立つ。

$$R_1(\theta, \rho) \equiv R_2(\theta + \alpha, \rho) \quad \forall \theta, \rho \quad (3)$$

$$R_1\left(\beta + \frac{\pi}{2}, \rho\right) \equiv R_3\left(\beta + \frac{\pi}{2}, \rho - d_0 \sin(\theta - \beta)\right) \quad \forall \rho \quad (4)$$

上式を用いると、ラドン変換法と同様に、物体の位置・姿勢を検出することができる。

本報告で提案するビジョンアルゴリズムは、演算量を減らすため $\rho = 0$ とし、物体の重心を原点として片側ラドン変換を行う。すなわち、

$$\tilde{R}(\theta) = R(\theta, 0) = \int_0^{\infty} g(\xi \cos \theta, \xi \sin \theta) d\xi \quad (5)$$

を計算する．片側ラドン変換においては， $R(\theta,0)$ と $R(\theta+\pi,0)$ の値は，一般に異なる．したがって，Fig.2-(b)に示すように θ と $\theta+\pi$ の区別ができる．また，物体によって片側ラドン変換の結果が異なる．この違いを利用して物体の種類を区別する．パターン画像の片側ラドン変換を $\tilde{R}_p(\theta)$ とし，入力画像の片側ラドン変換を $\tilde{R}_i(\theta)$ とすると，相関関数

$$S(\tau) = \int_0^{2\pi} \left\{ \tilde{R}_p(\theta + \tau) - \tilde{R}_i(\theta) \right\}^2 d\theta \quad (6)$$

を計算し， $S(\tau)$ の最小値 $S_{\min} = S(\tau_{\min})$ を求める．最小値 $S_{\min}$ が閾値以内であれば，入力画像はパターン画像の形状に一致すると判定する．このとき $\tau_{\min}$ が物体の姿勢である．

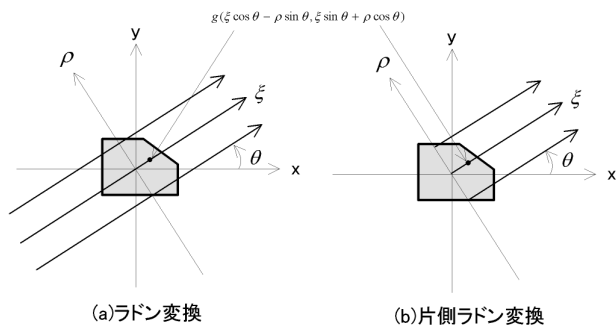


Fig. 1 ラドン変換と片側ラドン変換における積分経路

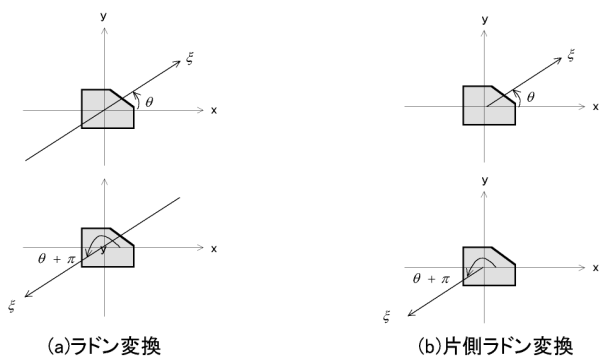


Fig. 2 $\rho = 0$ における積分経路

3. 実験

2節で述べたビジョンアルゴリズムを評価する．コントローラに自作 PC/AT 互換機(PentiumIII 450MHz)を選択し，OS はLinuxを選択した．コンパイラには，GCC Ver2.95.2を使用した．本報告では，平面運動する物体の姿勢認識を評価する．実験に使用した物体は，正三角形，正方形，円形の3種類である．物体認識においては，はじめに正三角形とのパターンマッチングを行う．次に正方形とのパターンマッチングを行う．正三角形，正方形ともにマッチングしないときには，円形と判断する．実験に使用する画像のサイズは幅 320×高さ 240で，カラーはグレースケールである．まず，画像データを切り出し，背景の余分な部分を除去する．次に，切り出した画像の2値化重心を計算する．重心を中心として64×64の画像を切り出す．この画像に対して，片側ラドン変換を計算する．このときの片側ラドン変換の積分分割数は32である．これらの画像処理時間をTable.1に示す．また，Fig.3の画像

を使用して，パターン画像と入力画像1，入力画像2の相関演算を行い， $S_{\min}$ と $\tau_{\min}$ の値を調べた．その結果をTable.2に示す．以上の実験により，提案するアルゴリズムで3種類の物体をビデオフレームレートで認識できることが示された．

Table. 1 画像処理時間

処理名	処理時間[msec]
画像データの切り出し	約 2.2
2 値化重心計算	約 1.7
ラドン変換	約 8.0
1 種類のパターンマッチング	約 3.6

Table. 2 相関関数の最小値

	$S_{\min}$	$\tau_{\min}$
入力画像 1	607×10^3	45
入力画像 2	478×10^6	

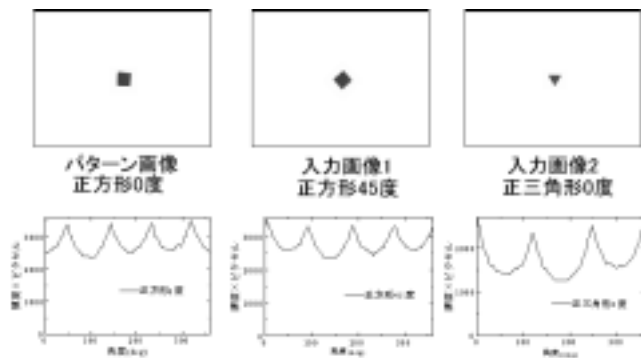


Fig. 3 パターンマッチングに用いる画像

4. おわりに

本報告では，片側ラドン変換を用いた物体認識をリアルタイムで行った．この実験により，リアルタイムで3種類の物体認識を行うことができた．しかし，OSからの割り込みによって画像処理の周期が乱れるため，ART-Linuxを導入しリアルタイム性を高める予定である．また，形状が同じで模様の違う物体の認識評価や1自由度ソータを用いた物体の仕分けを行う予定である．

参考文献

- [1] 尾崎，谷口，画像処理-その基礎から応用まで(第2版)，1983，152-205，240-243，共立出版
- [2] 徳本，斎藤，平井，変形推移グラフを用いたレオロジー物体の変形，ロボティクス・メカトロニクス講演会 '00 講演論文集 CD-ROM，2000
- [3] 大西，鈴木，有本，ハフおよびフーリエ変換を用いた拡大・回転・平行移動検出法の部品位置決めへの応用，日本ロボット学会誌，Vol.16，No.2，pp.232~240，1998