

リアルタイムビジョンのための片側ラドン変換法の並列化

座光寺正和 平井慎一 (立命館大学)

Parallel Processing of One-sided Radon Transform for Real-time Vision

*Masakazu ZAKOJI, Shinichi HIRAI(Ritsumeikan University)

Abstract—In this paper we will present parallel processing of one-sided Radon transform for real-time vision. One-sided Radon transform can detect the position and the orientation of a planar motion object. Unfortunately it takes much computation time and its real-time implementation is difficult. One-side Radon transform, however, has high parallel processing ability and its implementation on FPGA's is expected to reduce the computation time. In this paper, we will develop a parallel algorithm of one-side Radon transform.

Key Words: vision algorithms, one-sided Radon transform, parallel processing

1. はじめに

ビジョンシステムにおいてビデオ信号をその入力とするとき，リアルタイム性を保つためには画像処理を33ms（ビデオフレームレート）以内に行わなければならない．その画像処理をPCで行わせる場合，複雑な処理や計算量の多い処理を行わせようとするとこの制限を越えてしまう場合が多い．

ビデオフレームレート内での処理が難しいビジョンアルゴリズムに片側ラドン変換がある．このアルゴリズムでは物体の並進移動と回転を検出できるという特長を有する一方，計算量が多くビデオフレームレートでの処理が困難であるという欠点を有する．しかしアルゴリズムの並列性は高いためハードウェア化することによりビデオフレームレート内で処理できる可能性が高い¹⁾．そこで，本報告ではハードウェア化に適した片側ラドン変換の並列化アルゴリズムを構築する．

2. 片側ラドン変換

片側ラドン変換はラドン変換の積分路を $[0, \infty]$ に変更したものであり，入力画像を $g(x, y)$ とすると，Fig.1 に示すように片側ラドン変換 $R(\theta, \rho)$ は次式で与えられる．

$$R(\theta, \rho) = \int_0^{\infty} g(\xi \cos \theta + \rho \sin \theta, \xi \sin \theta + \rho \cos \theta) d\xi \quad (1)$$

ここで θ は x 軸からの角度であり， ρ は原点からの距離である．この $R(\theta, \rho)$ とあらかじめ計算しておいたテンプレート $R_0(\theta, \rho)$ との2次元マッチングを行うことにより物体の並進移動と回転の検出が可能となる²⁾．

3. アルゴリズムの並列化

3.1 ハードウェア化の問題点

片側ラドン変換を並列化するには，(1) 式の計算を行うモジュールを複数個作れば良い．(1) 式の変換アルゴリズムでは θ, ρ, ξ から x, y を計算するため，Fig.2 に示すように $g(x, y)$ へのアクセスがランダムになる．通常，一枚の画像は画面隅から一列ずつスキャンして，その順番どおりにメモリに蓄えられる．そのため，(1) 式による変換では，Fig.3 に示すように並列化したモジュール

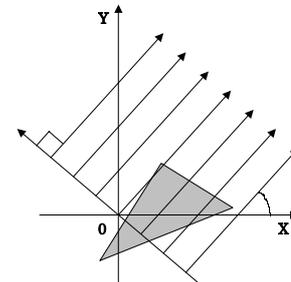


Fig.1 One-sided Radon transform

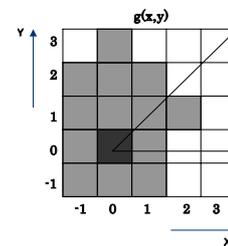


Fig.2 Order of memory access

ルが個々にスキャンの順番に関係なく画像メモリにアクセスを行うので，効率が悪く配線も複雑になってしまう．またランダムアクセスということは，一枚の画像 $g(x, y)$ がすべて揃ってからでないで変換を始められないということであり，その間モジュールは計算できずにただ待つだけになってしまう．

以上の問題を解決するためには $g(x, y)$ へ順次アクセスしながら片側ラドン変換が行えるようにアルゴリズムを変更する必要がある．また，順次アクセスしながら変換が行えると，Fig.4 に示すように次々に入ってくる画像データをメモリに保存しておく必要がなくなるので，画像用のメモリが不要になる．このことはハードウェア化にとって大きなメリットである．

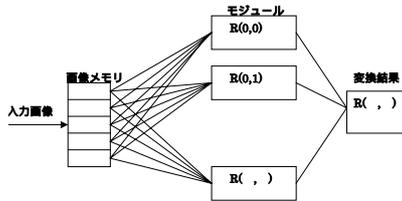


Fig.3 Parallel processing with image memory

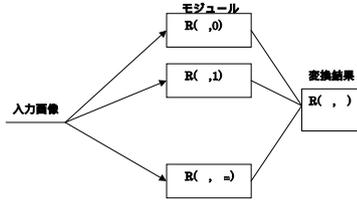


Fig.4 Parallel processing without image memory

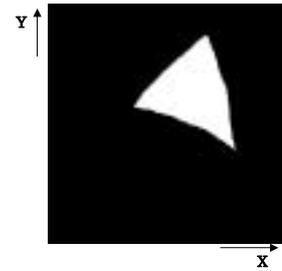


Fig.5 Input image

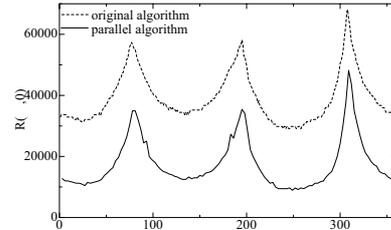


Fig.6 Transformed image($\rho = 0$)

3.2 順次アクセスによる変換方法

前節の問題解決のため $R(\theta, \rho)$ を $g(x, y)$ に順次アクセスしながら変換できるようにアルゴリズムを変更する。(1)式の $R(\theta, \rho)$ と $g(x, y)$ の関係から次の式が導かれる。

$$x = \xi \cos \theta - \rho \sin \theta \quad (2)$$

$$y = \xi \sin \theta + \rho \cos \theta \quad (3)$$

これを ρ と ξ について解くと

$$\rho = y \cos \theta - x \sin \theta \quad (4)$$

$$\xi = x \cos \theta + y \sin \theta \quad (5)$$

となり、 x, y, θ から ρ, ξ を計算し、 ξ の正負により $R(\theta, \rho)$ に画素値 $g(x, y)$ を投票するかどうか判断する。すなわち、 $\xi \geq 0$ のとき $R(\theta, \rho)$ の値を画素値 $g(x, y)$ だけ増やす。 $\xi < 0$ のときは何もしない。この手法により順次アクセスによる変換方法が実現できると考える。

4. 実験

PC 上でプログラムにより仮想的に順次アクセスによる変換方法を実現し、変換結果を比較する。入力画像には Fig.5 を用いる。画像サイズは $300 \times 300(\text{pixel})$ である。

4.1 $\rho = 0$ での変換

片側ラドン変換の中には、物体の重心を変換の原点とし $\rho = 0$ についてのみ変換を行うアルゴリズムがある。そこで、まず $\rho = 0$ の場合について順次アクセスによる変換を行う。

(4) 式において $\rho = 0$ とすると、 $\theta = \tan^{-1}(y/x)$ となる。これは画素と原点を結ぶ直線が x 軸となす角度 θ を求めることになる。 θ の分解能は 360 とした。Fig.6 が変換結果である。(1) 式での変換と比べると $R(\theta, 0)$ のスケールが小さくなっているだけで、認識には問題ない変換結果が得られていることがわかる。

4.2 $\rho = 0$ 以外への拡張

$\rho = 0$ の変換結果よりオリジナルと同じ変換結果が得られることがわかったので同じ方法で $\rho = 0$ 以外へ拡張する。3.2 節で述べた手法で $R(\theta, \rho)$ を計算する。変換結果を Fig.7 に示す。これによると、特定の ρ で $R(\theta, \rho)$ の値にピークがでる。しかし、認識には問題ない変換結果が得られている。

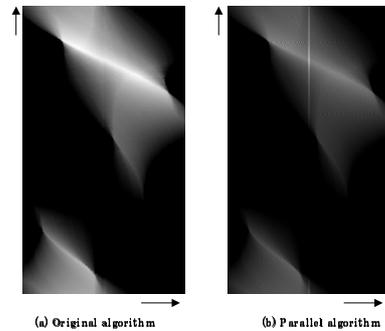


Fig.7 Transformed image

5. おわりに

PC 上に実現したプログラムで順次アクセスによる変換が可能であることは示された。今後は FPGA への実装を行い、リアルタイムビジョンを実現する。

参考文献

- 1) 増淵・平井: ロボット搭載用高速高機能ビジョンモジュールの研究; 第 18 回日本ロボット学会学術講演会予稿集, Vol.2, pp.649-650, (2000)
- 2) 坪井・平井: 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物体計測; 第 18 回日本ロボット学会学術講演会予稿集, Vol.2, pp.645-646, (2000)