棒状図形の傾き検出のラン符号による高速化の一手法

8Ｃ－6

古賀 昌史*　中島和樹*　丸川勝美*　嶋 好博*　藤澤 浩道**

*(株)日立製作所中央研究所　**(株)日立製作所ソフトウエア開発本部

1．はじめに

エッジ検出等により得られた棒状の図形の位置，傾きを高速に検出する技術が景観理解，工業用画像処理，文書画像処理等において必要となっている。

棒状の二値画像の図形の傾きを算出する方式としては，(a) 端点あてはめ法，(b) ハフ変換による方式[1]，(c) 最小二乗法による直線近似方式[2]等が知られている。このうち，(a)は使用する記憶容量，処理量が小さいが，精度は低い。(b)は，多くの記憶容量，処理量が必要という欠点がある。(c) は，精度が高く，必要な記憶容量が小さいが，処理量が多く上述の様な用途に用いるには速度が十分でない。

ここでは，画像をラン符号化して計算量を削減し，最小二乗法による傾き検出を高速化する方式を提案する。

2．黒画素図形傾き検出の技術課題

棒状図形傾き検出においては，精度と処理速度が大きな課題である。

エッジ検出などで得られる二値画像上の図形には通常，輪郭に鋸状の凹凸があり，太さも一定でなく，精度良く傾きを求めることは困難である。しかし最小二乗法による直線近似を用いると，±0.15rad. の範囲で誤差 1.0×10^{-3} rad. 以内で精度良く傾きを検出することができる[2]。

従来の最小二乗法は１画素毎に座標の二乗和や積和を求める必要があるため，多くの処理時間が必要であった。ここでは従来方式を改良し，処理時間を１／１０以下にすることを目標とする。

3．傾き検出方式の原理

3．1　従来方式（画素単位計算方式）

図１（ａ）に示すように二値画像上の棒状図形 f をもっともよく近似する直線 l，すなわち

$$y = a\cdot x + b \qquad \cdots\cdots\cdots \quad (1)$$

の傾き a を，f の傾きであると定義する。

最小二乗法による直線近似では，式(2)に示す各画素の座標と直線との距離の二乗和の近似値S を最小にするa，bの値を求める。

$$S = \sum_{i=1}^{M}\left(a\cdot x_i + b - y_i\right)^2$$

（(x_i, y_i) は図形fに含まれる全ての画素。$0\leq i\leq M$ ）　……… (2)

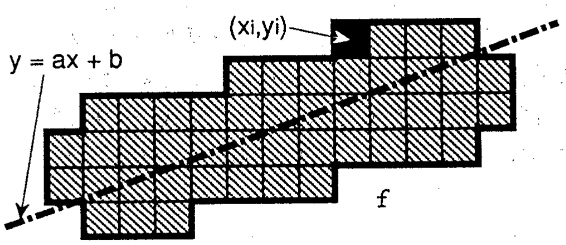
ここで， p, q, r, s, t を

$$p=\sum_{i=1}^{M}x_i^2,\ q=\sum_{i=1}^{M}x_i,\ r=M,\ s=\sum_{i=1}^{M}x_i\cdot y_i,\ t=\sum_{i=1}^{M}y_i \qquad \cdots\cdots\cdots \quad (3)$$

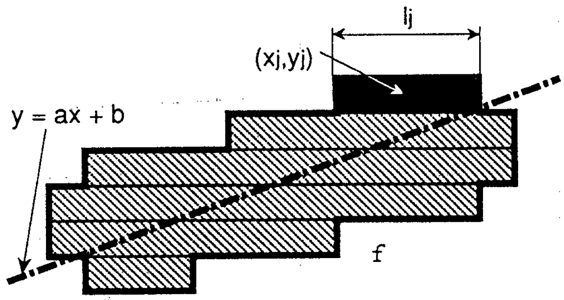
とすると，式(2)の S を最小にするa, b の値は，

$$a=\frac{r\cdot s-q\cdot t}{r\cdot p-q^2}\ ,\quad b=\frac{p\cdot t-q\cdot s}{r\cdot p-q^2} \qquad \cdots\cdots\cdots \quad (4)$$

の式で求めることができる。従来方式では，式(3)の計算を画素単位に行なっているため，多くの処理時間が必要であった。



（ａ）画素データによる直線近似



（ｂ）ランによる直線近似

図１　二値画像上の図形の直線近似

A Fast Skew Detection Method of Bar-Shaped Figure by Run Coding,
Masashi KOGA, Kazuki NAKASHIMA,
Katsumi MARUKAWA, Yoshihiro SHIMA,
Hiromichi FUJISAWA
Hitachi, Ltd.

3．2　ラン符号化利用方式

二値画像上の図形は図１（ｂ）に示すように，X方向の画素の並び（ラン）の集合として表現できる。ランによる二値画像の符号化には様々な形式があるが，ここでは，各ランを中点の座標と，ラン長の組（x，y，l）で表現する。

棒状図形に含まれるｊ番目のラン（x_j，y_j，l_j）毎に p_j, q_j, r_j, s_j, t_j を以下の式で定義する。

$$
\left\{
\begin{aligned}
p_j &= \sum_{x=x_j-l_j/2}^{x_j+l_j/2} x^2 = \left(x_j^2+\left(l_j^2-1\right)/12\right)\cdot l_j \\
q_j &= \sum_{x=x_j-l_j/2}^{x_j+l_j/2} x = x_j\cdot l_j \\
r_j &= l_j \\
s_j &= \sum_{x=x_j-l_j/2}^{x_j+l_j/2} x\cdot y = x_j\cdot y_j\cdot l_j \\
t_j &= \sum_{x=x_j-l_j/2}^{x_j+l_j/2} y = y_j\cdot l_j
\end{aligned}
\right.
\qquad \cdots\cdots\cdots \quad (5)
$$

ここで，

$$p=\sum_{j=1}^{N}p_j,\quad q=\sum_{j=1}^{N}q_j,\quad r=\sum_{j=1}^{N}r_j,$$

$$s=\sum_{j=1}^{N}s_j,\quad t=\sum_{j=1}^{N}t_j$$

（Nはfに含まれるランの総数）　　　　………　(6)

の p, q, r, s, t の各式に式(5)を代入し，この結果を式(4)に代入すると，棒状図形の傾き a を求めることができる。一般に二値画像中のランの数は黒画素の数の1/10から1/100であるため，式(5)の計算回数は従来方式における式(3)の計算より大幅に少なくなり，処理の高速化が図れる。

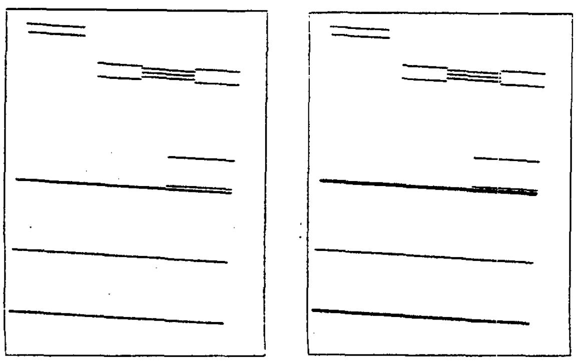


図２　棒状図形の例（サンプル２，傾き約 0.5rad.）

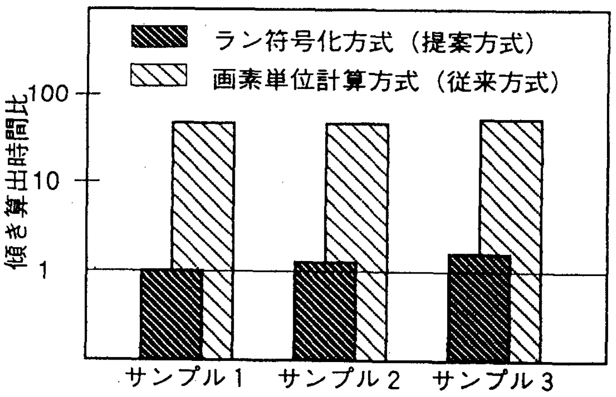


図３　傾き算出時間の比較

4．実験結果

本方式の処理速度を評価するため，二値画像棒状図形の傾き算出実験を行った。サンプルは３つの二値画像(864×1136画素)で，各サンプル中には平行な棒状図形が１４個存在する。各サンプル中の棒状図形の傾きはそれぞれ約 0rad., 0.05rad., 0.1rad.である。各サンプル中の図形に対し，傾きを最小二乗法で算出し，画素単位に積和等を算出する方式(従来方式)と，ラン符号より算出する方式(提案方式)で処理時間を比較する。処理は，ワークステーション(68030, 25MHz)上で，Ｃ言語で記述したプログラムにより実行した。

図２（ａ）はサンプル中の棒状図形の例である。図２（ｂ）は，最小二乗方式で算出した傾きより求めた棒状図形の外接矩形である。

図３はサンプル毎に傾き算出に要した処理時間の比較結果である。各サンプルの処理時間は，新方式でのサンプル１の処理時間に対する比で示している。新方式により，従来方式に対し１０倍以上高速に傾きが検出することができている。

5．まとめ

二値画像中の棒状図形の傾きを算出する方式について検討した。ラン符号を利用することにより，大幅に傾き検出を高速化できることがわかった。本方式は景観理解，工業用画像認識，文書画像処理等に応用できると考えている。

6．参考文献

[1]　松山他 "Hough 変換とパターンマッチング" 情報処理 Vol. 30, No. 9, pp. 1035 - 1046,　平成１年
[2]　古賀他 "二値画像棒状図形の傾き検出の一手法" 信学全, 1992秋, D-218, pp. 6-220,　平成４年
[3] Y. Nakano, "An algorithm for skew normalization of document image" Proc. of the 10th International Conf. on Pattern Recognition, vol.2, pp.8-13, 1990