

7J-2

対数表現数値システムによる 画像の幾何学的変換

黒河富夫

愛知工業大学 経営工学科

1. はじめに

コンピュータの数値演算には固定小数点と浮動小数点の演算が圧倒的である。しかし第三の数値演算法として、近年、対数表現数値システム(Logarithmic Number System:LNS)による演算法(対数表現数値演算法:Logarithmic Arithmetic:LA)が注目され研究されて来た⁽¹⁾⁻⁽⁴⁾。

2. 対数表現数値システム

LNSでは、数は(1)のように現される。

$$s d_1 d_2 \dots d_m \cdot 1_1 1_2 \dots 1_n \quad (1)$$

ここで、 $s, d_1, d_2, \dots, d_m$ は0または1で s は数値の符号、 d -部と 1 -部は2進数で数値の指数を表す。 d -部と 1 -部の間の点は暗黙の小数点、従って、 1 -部は指数の小数部である。指数の底も暗黙の値で、1より大きい数 a である。従って、(1)は

$$\pm a^{d\text{-part} \cdot 1\text{-part}} \quad (2)$$

を表す。

この数値システムでは乗算と除算は指数部の固定小数点の加算または減算でできるから非常に高速であり演算誤差もない。加算と減算はルックアップ・テーブルを使用すれば高速にできる。例えば、 a^x と a^y の加算は

$$a^z = a^x + a^y = a^x (1 + a^{y-x}),$$

2を底とする対数をとると、

$$z = x + \log_a (1 + a^{y-x}),$$

$\log_a (1 + a^{y-x})$ は $y-x$ をアドレスとするルックアップ・テーブルにより用意しておけば、簡単にできる⁽¹⁾。減算も同様である。これをチップ化し、同条件の浮動小数点演算と比較すると、加算で1:1、乗算で5:1、平方根等で20:1等、超高速化され⁽³⁾、また演算誤差も非常に小さくなる⁽⁴⁾と報告されている。

3. 画像の幾何学的変換への応用

このLNSは、従来、信号処理、特にデジタル・フィルタリングを主眼に研究開発されてきた。従って、画像処理やCG、特に幾何学的変換で必要なアドレス計算に使用することは殆ど試みられなかった。信号処理では図1で示すような形で使用されるが、幾何学変換等のアドレス処理では、アドレスという整数で表されている数値をわざわざ誤差を出して、対数に変換し、また、係数も対数に変換しておいて、それらをLAで計算し、必要に応じて、元の整数アドレスに変換し直すことにより処理を行う。アドレス整数からLNS、LNSからアドレス整数、

双方の変換はルックアップ・テーブルにより高速に行える。

図2(c)は上記の方法による画像のアフィン変換のシミュレーション例である($m=5, n=9, a=2$ の2バイト数値を使用)。(b)は64ビットの浮動小数点演算によるものであるが、差異は認められない。(b),(c)どちらも最近傍法によった。(a)は原画像で101*101、(b)と(c)は121*121の大きさである。



ALA: Analog Logarithmic Amplifier, AD: A/D Converter, LAP: Logarithmic Arithmetic Processor, DA: D/A Converter, AEA: Analog Exponential Amplifier

図1 対数表現数値システム信号処理の構成

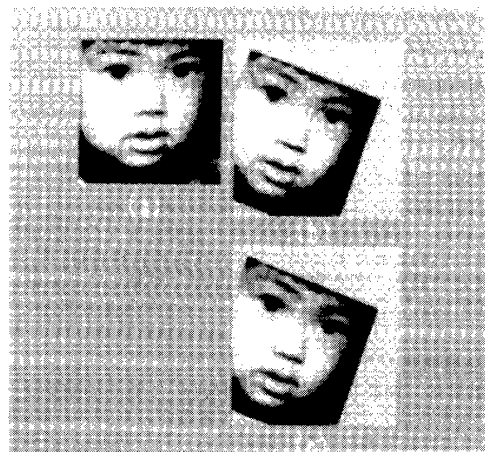


図2 アフィン変換の例

参考文献: (1)Kingsbury et al., "Digital Filtering Using Logarithmic Arithmetic", Electron. Lett. 7, 56-58, 1971 (2)Lamaire et al., "Performance of Digital Linear Regulators Which Use Logarithmic Arithmetic", IEEE AC-31, 394-400, 1986 (3)Taylor et al., "A 20 Bit Logarithmic Number System Processor", IEEE C-37, 190-200, 1988 (4)Kurokawa et al., "Error Analysis of Recursive Digital Filters Implemented with Logarithmic Number Systems", IEEE ASSP-28, 706-715, 1980