

FPGA 実装のための GVF Snakes の並列化

○ 山本哲也, 平井慎一 (立命館大学)

Parallel GVF Snakes to implement on FPGA

○ Tetsuya Yamamoto, Shinichi Hirai (Ritsumeikan Univ.)

Abstract: This paper describes the design of a parallel computing circuit for GVF Snakes. GVF Snakes can extract the boundary of an object. But, requires much computation time. We design a circuit to perform GVF Snakes on an FPGA.

1 始めに

動的輪郭モデル (Snakes) は, ある定義したエネルギー (内部エネルギーと画像エネルギー) を最小化することによって輪郭を抽出・追跡する手法である。しかし, 一般的な Snakes では, 凹みに対応させることが難しい。凹みに対応できる動的輪郭モデルとして, GVF Snakes が提案されている。Snakes, GVF Snakes は, 計算量が多く, PC 上では処理に時間がかかってしまうという欠点がある。そこで, 本研究では, GVF Snakes を FPGA 上に実装し, 高速の GVF Snakes を実現する。本報告書では, FPGA 上に実装するための GVF Snakes の並列化について述べる。

2 GVF Snakes

GVF Snakes[1] では, 原画像の勾配画像を求め, 求めた勾配画像に対して Snakes を適用する。すなわち, エッジ付近から離れるに従って滑らかに輝度値が小さくなるような GVF 画像 $V_{i,j}$ を求める。以下の式で与えられる。

$$V_{i,j} = \sqrt{(u_{i,j})^2 + (v_{i,j})^2} \quad (1)$$

ここで, $(u_{i,j}, v_{i,j})$ は, 格子点 (i, j) における勾配ベクトルであり, 以下の計算を繰り返し行うことにより求められる。

$$u_{i,j}^{n+1} = (1 - b_{i,j}\Delta t)u_{i,j}^n + c_{i,j}^1\Delta t + r(u_{i+1,j}^n + u_{i,j+1}^n + u_{i-1,j}^n + u_{i,j-1}^n - 4u_{i,j}^n) \quad (2)$$

$$v_{i,j}^{n+1} = (1 - b_{i,j}\Delta t)v_{i,j}^n + c_{i,j}^2\Delta t + r(v_{i+1,j}^n + v_{i,j+1}^n + v_{i-1,j}^n + v_{i,j-1}^n - 4v_{i,j}^n) \quad (3)$$

ただし,

$$r = \frac{\mu\Delta t}{\Delta x\Delta y} \quad (4)$$

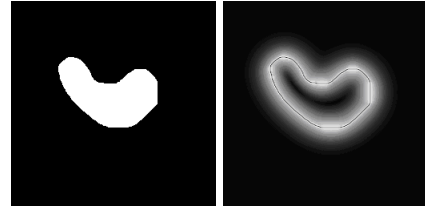
である。

エッジ画像にソーベルフィルタを施した結果を $b_{i,j}$ で表す。また, $b_{i,j}$ に縦ソーベルフィルタをかけた結果を $c_{i,j}^1$, $b_{i,j}$ に横ソーベルフィルタをかけた結果を $c_{i,j}^2$ で表す。

求められた GVF 画像の例を Fig.1 に示す。1000 回繰り返し計算を行った結果である。Fig.1 から輝度値がエッジを中心に滑らかに広がっていることがわかる。

3 移動処理方法

GVF Snakes では, 輪郭を有限個の離散点の閉曲線で表す。エネルギーが最小になる離散点を見つける。本研究では, 現在の離散点とその 8 近傍のエネルギーを計算して, エネルギーが最小となる点を新たな離散点とする。



(a) original image (b) GVF image

Fig. 1: Example of GVF image

3.1 エネルギー最小化

離散点 P_s に依存するエネルギーは次式で表せる。

$$E_s^{snake} = E_s^{int} + E_s^{img} \quad (5)$$

ここで, E_s^{int} は内部エネルギー, E_s^{img} は画像エネルギーを表す。 E_s^{snake} が最小になるように, 離散点 P_s を移動させる。

3.2 内部エネルギー

内部エネルギーは Snakes の形状の滑らかさを拘束しながら収縮させるエネルギーであり, このエネルギーを最小化させることにより Snakes を収縮させ, より滑らかな形状に変形させる。ある離散点 P_s に依存する内部エネルギー E_s^{int} を, 次式で表す。

$$E_s^{int} = d_s + \theta_s \quad (6)$$

ここで, d_s と θ_s は, 以下の 2 式で与えられる。

$$d_s = \frac{1}{2} \left(\left| \overrightarrow{P_s P_{s+1}} \right|^2 + \left| \overrightarrow{P_{s-1} P_s} \right|^2 \right) \quad (7)$$

$$\theta_s = 1 - \frac{\overrightarrow{P_{s-1} P_s} \cdot \overrightarrow{P_s P_{s+1}}}{\left| \overrightarrow{P_{s-1} P_s} \right| \left| \overrightarrow{P_s P_{s+1}} \right|} \quad (8)$$

3.3 画像エネルギー

画像エネルギーは画像の特徴を表すエネルギーであり, このエネルギーを最小化させることにより Snakes を対象物のエッジに張り付くように変形させる。ある離散点 P_s に依存する画像エネルギー E_s^{img} を, 次式で表す。

$$E_s^{img} = I_{s,s+1} + I_{s,s-1} \quad (9)$$

ここで, $I_{s,s+1}$ は離散点 P_s と P_{s+1} をつなぐ線分上の GVF 画像の輝度値の総和である。

4 並列化

FPGA 上に実装するための並列化について述べる。GVF 画像の計算, 内部エネルギーの計算, 画像エネルギーの計算を並列化する。

4.1 GVF 画像の計算

GVF 画像の計算は, Fig.2 のように並列化が可能である. ソーベルフィルタによるエッジ画像を $f_{i,j}$ で表す. ここで, $b_{i,j}$ はエッジ画像にソーベルフィルタをかけた結果であり, $c_{i,j}^1$ は $b_{i,j}$ に縦ソーベルフィルタをかけた結果であり, $c_{i,j}^2$ は $b_{i,j}$ に横ソーベルフィルタをかけた結果である. 勾配ベクトルの要素 $u_{i,j}$ と $v_{i,j}$ は, それぞれ並列に処理できる. 導出した $u_{i,j}$ と $v_{i,j}$ から GVF 画像 $V_{i,j}$ を出力する.

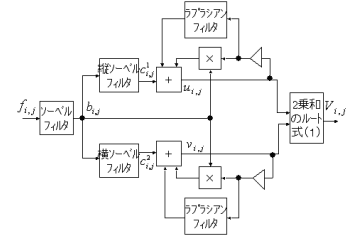


Fig. 2: Parallel computing of GVF image

4.2 内部エネルギーの計算

内部エネルギーの計算は, Fig.3 のように並列化が可能である. $P_s = (i_s, j_s)$ の移動を考えた場合, P_{s-1}, P_s の距離, P_s, P_{s+1} の距離, $\angle P_{s-1}P_sP_{s+1}$ を並列に計算することができる. 重み係数をかけて, それぞれを加算して P_s の内部エネルギー E_s^{int} を出力する. また, 離散点 P_{s-1} と P_s の距離の計算は Fig.4 のように, 角度 $\angle P_{s-1}P_sP_{s+1}$ の計算は Fig.5 のように並列化が可能である.

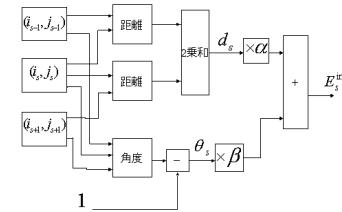


Fig. 3: Parallel computing of internal energy

4.3 画像エネルギーの計算

画像エネルギーの計算は, Fig.6 のように並列化が可能である. まず, P_s と P_{s-1} までをつなぐ線分上の格子点と P_s と P_{s+1} までをつなぐ線分上の格子点を Bresenham のアルゴリズム [2] を用いてそれぞれ生成し, 生成した格子点を基に GVF 画像の輝度値を得て, 足し合わせる. P_s から P_{s-1} の足された輝度値と P_s から P_{s+1} の足された輝度値を足して P_s の画像エネルギー E_s^{img} を出力する.

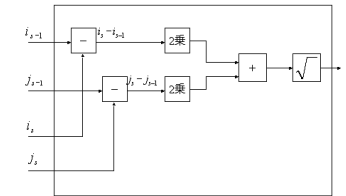


Fig. 4: Circuit to compute distance

4.4 全体の構成

離散点 P_s に依存する全エネルギー E_s^{snake} の計算は, Fig.7 のようになる. 画像エネルギーの計算は, GVF 画像の計算の後に実行する. 内部エネルギーの計算と画像エネルギーの計算は並列に処理することが可能である.

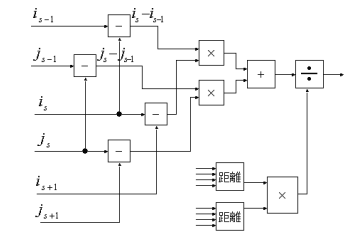


Fig. 5: Circuit to compute angle

5 終わりに

本報告では, GVF Snakes を FPGA 上に実装するための並列化について述べた. 今後は FPGA 上に実装し, 実際に考えた並列化でリアルタイムによる抽出ができるか検証を行う.

参考文献

- [1] Chenyang Xu and Jerry L. Prince: "Snakes, Shapes, and Gradient Vector Flow"; IEEE Transactions on Image Processing, 359-369, 1998 .
- [2] Colin Flangan: "The Bresenham Line-Drawing Algorithm"; <http://www.cs.helsinki.fi/group/goa/mallinnus/lines/bresenh.html>, URL.

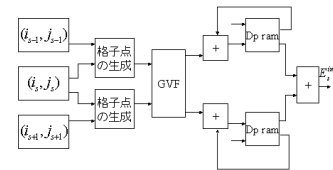


Fig. 6: Parallel computing of image energy

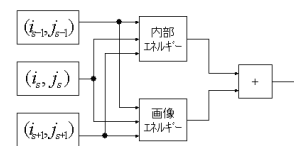


Fig. 7: Composition of all