

## Active Tubes による動画解析のためのロバストなアルゴリズム

4F-3

古川 亮, 今井 正和, 烏野 武  
奈良先端科学技術大学院大学

## 1 はじめに

連続画像から物体の運動を抽出することは、重要かつ困難な問題である。我々は [1] で、非剛体物体の動きを抽出するために Active Tubes を提案した。Active Tubes は時空間画像 [2] (連続画像を積み重ねた三次元画像) を利用し, Snakes [3] と同様なエネルギー最小化の手法を用いて物体の抽出を行う。[1] ではエネルギー最小化アルゴリズムとして Greedy Algorithm [4] を用いた。本研究では, Greedy Algorithm がノイズの影響を受け易いことを指摘し, 新しく, Greedy Algorithm と同程度に高速でノイズに対してより頑健な Greedy Algorithm の拡張アルゴリズムを提案する。

## 2 Greedy Algorithm と捕獲

Active Tubes では物体の運動解析に時空間画像を用いる。物体は時空間画像中で柱状の領域をなし, 同物体は時間軸に並行でない側面を持つ。Active Tubes は時空間画像中で管状の形状をしており, 目的とする領域の表面に収束させることができる。このことを利用して物体の運動を追跡することができる。

Active Tubes の表面上の点を  $v(s, t)$  と表す。ここで  $t$  は時間領域の,  $s$  は空間領域のパラメータである。Active Tubes のエネルギー関数  $E_{tube}^*$  を以下のように定義する。

$$E_{tube}^* = \int_T \int_0^1 \{E_{int}^*(v(s, t)) + E_{ext}^*(v(s, t))\} ds dt \quad (1)$$

$E_{int}^*$  を内部エネルギー,  $E_{ext}^*$  を外部エネルギーと呼ぶ。前者は Active Tubes の形状を保つ役割を, 後者は Active Tubes を目的の領域の表面に引き付ける役割を持つ。

計算機上にインプリメントする場合, Active Tubes は離散形式で表される。離散形式では, 各ノードは  $i$  と

$t$  の二つの添字で表される。  $E_{tube}^*$  の離散形式  $E_{tube}$  は,

$$E_{tube} = \sum_T \sum_I \{E_{int}(v_{i,t}) + E_{ext}(v_{i,t})\}. \quad (2)$$

のようになる。  $E_{int}^*$  の離散形式  $E_{int}$  は,

$$\begin{aligned} E_{int}(v_{i,t}) = & \alpha_{i,t} (|v_{i,t} - v_{i-1,t}| - \bar{d})^2 \\ & + \beta_{i,t} |v_{i-1,t} - 2v_{i-1,t} + v_{i+1,t}|^2 \\ & + \gamma_{i,t} (|v_{i,t+1} - v_{i,t}|^2 + |v_{i,t-1} - v_{i,t}|^2) \\ & + \delta_{i,t} |v_{i,t-1} - 2v_{i,t} + v_{i,t+1}|^2 \end{aligned} \quad (3)$$

となる。ここで  $\bar{d}$  はノード間の距離の平均を表す。  $\alpha_{i,t}$ ,  $\beta_{i,t}$ ,  $\gamma_{i,t}$ ,  $\delta_{i,t}$  はそれぞれ空間領域での連続性, 空間領域での滑らかさ, 時間領域での連続性, 時間領域での滑らかさを制御する重み係数である。

Active Tubes を目的領域の表面に引き付けるために,  $E_{ext}^*$  は, 時空間画像上のエッジの所で最小になるように定義しなければならない。  $E_{ext}^*$  の離散形式としては,

$$E_{ext}(v_{i,t}) = w_{edge} \left\{ - \left| \frac{\partial}{\partial x} I(x, y, t) \right|^2 - \left| \frac{\partial}{\partial y} I(x, y, t) \right|^2 \right\} \quad (4)$$

の形のものがよく使われる。ここで  $I(x, y, t)$  は時空間画像上で  $v_{s,t}$  のある場所の画素の輝度値であり,  $w_{edge}$  は重み係数である。

[1] において, 我々は計算量を減らすために Greedy Algorithm を Active Tubes の収束に用いた。 Greedy Algorithm では,  $E_{tube}$  を最小化するノードの位置を, 繰り返し計算によって求める。 Active Tubes の変形は以下の手順を繰り返すことで行う。まず  $v_{s,t}$  とその近傍について, その他の全てのノードが動かないと仮定した時の  $E_{tube}$  を計算する。次に, ノード  $v_{s,t}$  を近傍上で  $E_{tube}$  が最小となる位置に動かす。全てのノードに対して同様の処理を行い, これを一回の繰り返し計算とする。一回の繰り返し計算中で動いたノードの数はカウントされ, この数が一定値より小さい時, Active Tubes が収束したと判定する。このアルゴリズムの計算量は  $O(mn)$

( $n$  はノードの数,  $m$  は近傍の数) であり高速であるが, 以下に述べるようにノイズの影響を受けやすいという欠点がある.

画像にノイズがあると, その部分の勾配のノルムはしばしば非常に大きくなる. この場合, 式(4)から, その場所に外部エネルギーの極小値ができ, これが Active Tubes のノードを引き寄せる. この外部エネルギーの極小値を 'pit' と呼び, pit に Active Tubes のノードが引き付けられる現象を '捕獲' と呼ぶ.

### 3 Randomized Greedy Algorithm

捕獲の問題を解決するために, 我々は Randomized Greedy Algorithm(RGA) を考案した. RGA は simulated annealing の考え方を取り入れたものであり, ノードの選択を確率によって行う.

以下に RGA の手続きを示す. まず  $v_{s,t}$  とその近傍について, Greedy Algorithm と同様に  $E_{tube}$  を計算する. 次に, それらの近傍について優先度を定義し, それに基づいて確率関数を決定する. 優先度は,  $E_{tube}$  の小さい順に,  $0, 1, \dots$  と定義する. 確率関数は,  $a$  を確率因数,  $n(x, y, t)$  をその近傍の優先度として,

$$P(x, y, t) = (1 - a)a^{n(x, y, t)} \quad (5)$$

の形式をとる. ノードは  $P(x, y, t)$  の確率で座標  $(x, y, t)$  の位置に移動する. 以上の手順を全てのノードについて行うことを一回の繰り返し計算とし, これを反復する.

全ての近傍についての  $P(x, y, t)$  の和は  $N$  を近傍画素数として,

$$(1 - a) + (1 - a)a + \dots + (1 - a)a^{N-1} = 1 - a^N \quad (6)$$

となり,  $1$  より  $a^N$  だけ小さい. よって  $a^N$  の確率でノードの次の位置が決められないことがある. この場合, ノードは現在の位置にとどめる.

収束判定は次のように行う. 各々のノードについて, 過去  $m$  回の  $x$  座標,  $y$  座標の分散を計算し, 両方ともある定数以下なら, そのノードは安定しているとみなす. 安定したノードがノード全体の一定の割合を越えたら, Active Tubes は収束しているとみなせる.

RGA は Greedy Algorithm と同程度に高速なうえに, ノイズに対して Greedy Algorithm より頑健である.

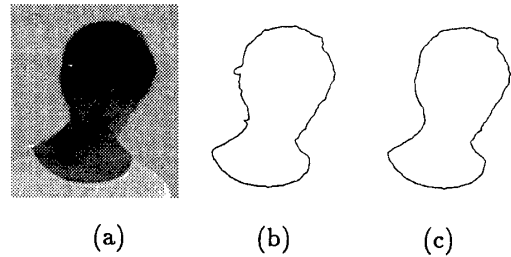


図 1: Greedy Algorithm と RGA の実験例

その理由は, Greedy Algorithm は決定論的アルゴリズムであるため, ノードが捕獲されるとそれが落とし穴から抜け出せなくなることがあるのに対し, RGA では確率的アルゴリズムであり捕獲されたノードであっても一定の確率でその状態から脱出できるためである.

RGA の性能を確かめるため, Greedy Algorithm と RGA とで実験を行った. その結果を図 1 に示す. (a) が原画像, (b) が Greedy Algorithm の結果, (c) が RGA の結果である. 反復回数はどちらも 20 回とした. Greedy Algorithm では幾つか捕獲が見られるが, RGA ではそれがなく分かる. また実行時間は Greedy Algorithm が 222 秒, RGA が 225 秒であり同程度である.

### 4 おわりに

本論文では, Active Tubes のアルゴリズムとして, Greedy Algorithm がノイズに弱いことを指摘した. これはノイズによって生じるエネルギーの極小状態から Active Tubes が脱出できなくなるためであった. また, simulated annealing の考え方を取り入れた Randomized Greedy Algorithm(RGA) を提案し, その有効性を実験により示した.

### 参考文献

- [1] Minoru Takahata, Masakazu Imai and Saburo Tsuji, "Determining Motion of Non-Rigid Objects by Active Tubes", *Proc. of 11th ICPR*, Vol.1, pp.647-651, 1992.
- [2] Masanobu Yamamoto, "Motion Analysis by Visualized Locus Method", *Tras. of Information Processing Society of Japan*, Vol. 22, No. 5, pp. 442-449, 1981. (In Japanese)
- [3] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active Contour Models", *International Journal of Computer Vision*, Vol.1, No.4, pp.321-331, 1988.
- [4] D. J. Williams and M. Shah, "A Fast Algorithm for Active Contours", *Proc. of Third ICCV*, pp.592-595, 1990.