

ファジー推論ニューラルネットワークを用いた 動画画像からの動オブジェクト抽出

桑原 修一 小原 仁 坂田 真人

秋田大学 工学資源学部 電気電子工学科

1. はじめに

動画画像から領域の動きを抽出するときには通常、ブロックマッチングにより動きベクトルを検出し、ブロック単位で動き予測を行っている[1]. この方法はブロックごとに操作を行うため、一つのオブジェクトであっても部分的に異なる方向を検出してしまふことが多い. 本研究では、動画画像中の移動するオブジェクトに対して FINN (Fuzzy Inference Neural Network) [2] を用いて動きベクトル予測を行い、それらを抽出することを目的とする.

2. 定義とモデル

2.1 動画画像

解析する動画画像は、静止画像を時間軸上に並べたものである. ここでは、固定されたカメラ位置から動いている物体を捉えた1シーンを対象とする. 動画画像のあるフレームについて、画素値や座標の近いものの集まりを領域とし、領域を類似な性質でまとめたものをオブジェクトと呼ぶことにする.

2.2 FINN のモデル

FINN のモデルを図1に示す.

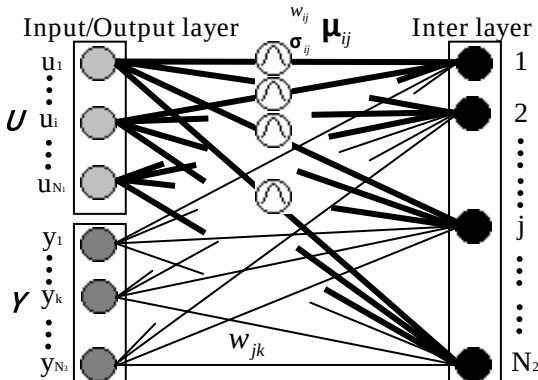


図1. FINN のモデル

入力変数の数を N_1 , 中間層のノードの数を N_2 , 出力変数の数を N_3 とする. 入力ベクトルを次式で表す.

$$U = (u_1, u_2, \dots, u_i, \dots, u_{N_1}) \quad (1)$$

入力層と中間層の間に配置されるメンバーシップ関数はベル型の次の関数である.

$$\mu_{ij} = \exp\left(-\frac{(u_i - w_{ij})^2}{\sigma_{ij}^2}\right), \quad j = 1, 2, \dots, N_2 \quad (2)$$

ここで, w_{ij} はベル型関数の中心値, σ_{ij} は分散値. 中間層の各ノードに入力されるあいまいさ μ_{ij} の最小

値を, 適応度 ρ_j として求める.

$$\rho_j = \min_j [\mu_{1j}, \mu_{2j}, \dots, \mu_{ij}, \dots, \mu_{N_1j}] \quad (3)$$

そして後件部定数 w_{jk} から, 次式より推定値 $\hat{y}_k$ を得る.

$$\hat{y}_k = \frac{\sum_j^{N_2} w_{jk} \rho_j}{\sum_j^{N_2} \rho_j}, \quad k = 1, 2, \dots, N_3 \quad (4)$$

2.3 FINN の学習

FINN の学習は, Kohonen のアルゴリズムによる自己組織化, 中間ノードの統合, LMS 学習から成る.

2.3.1 自己組織化

学習データ I は入出力層のベクトルを統合し,

$$I = (U, 0) + (0, Y) \quad (5)$$

とする. ここで, 出力ベクトルは,

$$Y = (y_1, y_2, \dots, y_k, \dots, y_{N_3}) \quad (6)$$

とし, 中間ノードへの重みベクトルは次式で表す.

$$W_j = (w_{1j}, w_{2j}, \dots, w_{ij}, \dots, w_{iN_1}, w_{j1}, w_{j2}, \dots, w_{jk}, \dots, w_{jN_3}) \quad (7)$$

計算手順はまず, ノード j への重みベクトル W_j と入力ベクトル I の距離を算出し, 最も距離の小さいノードを勝ちノード j^* とする. そして全ての重みベクトルを以下の式により更新する.

$$W_j(s+1) = W_j(s) + \epsilon_1(s) h(j, j^*, s) (I - W_j(s)) \quad (8)$$

ここで $\epsilon_1(s)$ は時間とともに減少する係数である. また, $h(j, j^*, s)$ は次式で表される.

$$h(j, j^*, s) = \exp\left(-\frac{d^2(j, j^*)}{\sigma^2(s)}\right) \quad (9)$$

ここで, $d(j, j^*)$ は重みを更新するノード j と勝ちノード j^* との距離, $\sigma(s)$ は時間とともに減少する係数.

2.2.2 中間ノードの統合

中間ノード間の距離が閾値 δ よりも小さければノードを統合する.

2.3.3 LMS 学習

Kohonen の自己組織化アルゴリズムにより, w_{ij} および w_{jk} が仮決定された. 次に, LMS (Least Mean Square) 学習アルゴリズムを用いて, メンバシップ関数の中心値 w_{ij} , 分散値 σ_{ij} , および後件部定数 w_{jk} の改善を行う. n 番目の入力に対する誤差 E_n は,

$$E_n = \frac{1}{2} \sum_k^{N_3} (y_k - \hat{y}_k)^2 \quad (10)$$

となり, 総誤差関数は,

$$E = \sum^n E_n \quad (11)$$

であるので、 E を最小にするように w_{ij} , σ_{ij} , および w_{jk} を順次更新する。

3. 処理アルゴリズム

図2に示す手順で操作を行う。

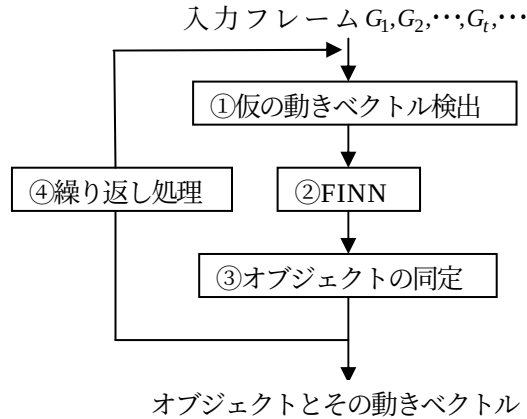


図2. 処理アルゴリズム

3. 1 仮の動きベクトル検出

t フレーム目と $t-1$ フレーム目の間の、仮の動きベクトル (d_x, d_y) を検出する。前フレームの情報が無い場合はブロックマッチング法を用いる。なお、検出した動きベクトルの確かさ p (確信度) を導入する。

3. 2 FINN

FINN の処理は図3のように行う。

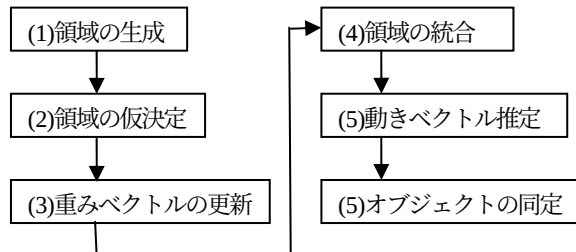


図3. FINN の処理の流れ

(1) 領域の生成

FINN のモデルにおける中間層の各ノードは、領域に対応する。ここでは、領域を生成する。各ノードの重みベクトルを、入力ベクトルの各要素に対応するようにランダムに生成する。

(2) 領域の仮決定

入力ベクトルと各中間ノードの重みベクトルとの距離を算出し、最も近いノードを勝ちノードとする。勝ちノードとなった領域を、入力画素を含む領域とする。

(3) 重みベクトルの更新

すべての重みベクトルを(8)式により更新する。

(4) 領域の統合

距離の近いノードを統合する。ただし、勝ちノードとなった回数の多いノード (大きな領域) に重みを持

たせる。

(5) 動きベクトル推定

まず、 n 番目の入力画素に対する適応度 ρ_{nj_1} を求め、(4)式から動きベクトル $(\hat{d}_x, \hat{d}_y)$ を推定する。

(6) LMS 学習

求めた動きベクトルの推定値を教師信号として w_{ij} , σ_{ij} , および w_{jk} の値を LMS 学習により更新する。

3. 3 オブジェクトの同定

入力した画素値がどのノードに属しているかの推定を行い、次いでオブジェクトを同定する。まず、 n 番目の入力画素に対する適応度 ρ_{nj_1} を求め、適応度の大きい順にそれぞれ、 ρ_{nj_1} ρ_{nj_2} とする。 j_1 および j_2 を、 n 番目の画素が属するオブジェクトの第一候補および第二候補とする。次に、縦軸に第一候補、横軸に第二候補のマップを作成し、二つの適応度が近ければ同一オブジェクトの候補としてマップ上に記録する。記録数の多かったノード同士を同じオブジェクトとみなす。そして、画素それぞれにノード番号を結び付けることでオブジェクトを同定する。

3. 4 繰り返し処理

$t-1$ 番目のフレーム G_{t-1} を利用する。まず、 G_{t-1} の各ピクセル点に関して、動きベクトルの推定値 $(\hat{d}_x, \hat{d}_y)$ だけ変位させた予測画像 G'_t を作成する。次に、 G'_t において検出されている領域を、 G_t の領域として前フレームとの動きベクトル検出を行い、仮の動きベクトルとして入力値に使用する。なお、画素を移動させたことにより生じた空白部分は、画素と領域が結びついていないので、ブロックマッチング法により仮の動きベクトルを求め、同座標における G_t の画素値とともに新しい領域としてノードに追加する。

4. 実験と結果

提案手法を PC に実装し、動画像を入力して出力を得た。単調な動画像では、はじめは動オブジェクトや背景が同じオブジェクトとして出力されたが、入力フレーム数を増やしていくと、抽出されるオブジェクトの形状が対象とするオブジェクトの形状に近づき、動きのない背景領域は大まかに抽出され、動きのあるオブジェクトは細かく抽出された。

実写の動画像入力に対して得られた結果は、動きのあるオブジェクトに関しては入力フレーム数を増やしていくことで元のオブジェクトに近い形で抽出されていった。しかし、背景領域も画素値の変化があるため細かいオブジェクトとして検出される部分があった。

5. まとめ

領域検出により細かい領域の検出を行い、ファジー推論によってそれらを統合することでオブジェクトを同定する方法を提案した。入力フレーム数を増やすことでノードの更新が進み、元のオブジェクトの形状に近づくことが分かった。

[参考文献]

- [1]電子情報通信学会編, 映像情報符号化pp121-141, オーム社出版局 (2001)
- [2]Takatoshi Nishina, Masafumi Hagiwara, 'Fuzzy inference neural network' Neurocomputing, pp223 -239(1997)