Spatio-temporal volume を用いた繰り返し処理のグラフカットによるビデオセグメンテーション

永橋　知行†　　藤吉　弘亘†　　金出　武雄††

† 中部大学大学院工学研究科 〒 487-8501 愛知県春日井市松本町 1200
†† カーネギーメロン大学 〒 15213 米国ペンシルバニア州ピッツバーグ市
E-mail: †{nagahashi,hf}@vision.cs.chubu.ac.jp, ††tk@cs.cmu.edu

あらまし　本稿では，Spatio-temporal volume を繰り返し処理による Graph Cuts を用いた高精度なビデオセグメンテーション法を提案する．入力された動画像に対し，時空間データに対して Mean Shift クラスタリングを行い Spatio-temporal volume を作成する．Mean Shift クラスタリングのバンド幅を変化させて作成した Spatio-temporal volume に対し，バンド幅が大きなものから Graph Cuts によりセグメンテーションを行う．セグメンテーション結果から空間的確率と色分布確率を計算し，それらを次の Graph Cuts の t-link に利用することで，大域的な情報を局所的な情報へ伝播させる．これらを繰り返し処理することにより，大域的なビデオセグメンテーションから段階的に局所的なビデオセグメンテーションを実現する．

キーワード　物体抽出，ビデオセグメンテーション，グラフカット

Video Segmentation Using Iterated Graph Cuts Based on Spatio-temporal Volume

Tomoyuki NAGAHASHI†, Hironobu FUJIYOSHI†, and Takeo KANADE††

† Dept. of Computer Science, Chubu Univ. Matsumoto 1200, Kasugai, Aichi, 487-8501 Japan
†† The Robotics Institute, Carnegie Mellon Univ. Pittsburgh Pennsylvania, 15213 USA
E-mail: †{nagahashi,hf}@vision.cs.chubu.ac.jp, ††tk@cs.cmu.edu

Abstract　We present a novel approach to video segmentation using iterated Graph Cuts based on Spatio-temporal volume. We use the Mean Shift clustering algorithm to build the spatio-temporal volumes with different bandwidth from the input video. We compute the prior probability obtained by the likelihood from a color histogram and a distance transform using the segmentation results from graph cuts in the previous process, and set the probability as the t-link of the graph for the next process. The proposed method can segment the regions of an object with a stepwise process from global to local segmentation by iterating the graph-cuts process with Mean Shift clustering using different bandwidth.

Key words　Foreground Extraction, Video Segmentation, Graph Cuts

1. はじめに

手持ちカメラで撮影された動画像から対象物体領域を時系列に抽出するビデオセグメンテーションは難しい問題である．ビデオセグメンテーションは動画像の合成や物体認識やジェスチャ認識などの前処理として利用されるため重要な技術である．

ビデオセグメンテーション手法として，Graph Cuts [1] [2] [3] [4] によるセグメンテーションが提案されている．Graph Cuts によるセグメンテーションは，ラベルに対するエネルギーを最小化させることで問題として解く手法で，Boykov らにより Interactive Graph Cuts [2] [3] が提案されている．Interactive Graph Cuts では，ユーザが与えた正解ラベルと入力画像からグラフを作成し，minimum cut/maximum flow algorithm を用いることで，エネルギー関数の最小化を行う．Graph Cuts によるセグメンテーションの拡張手法として，永橋らは，平滑化度合を変化させながら繰り返しグラフカットセグメンテーションを実行することで，より高精度なセグメンテーション手法を提案している [5]．Graph Cuts によるセグメンテーションは，動画像を時系列のボリュームデータとみなすことで，画像セグメンテーションと同様の枠組みでビデオセグメンテーショ

ンに応用することができる．しかし，動画像中の各ピクセルをノードとしてグラフを作成するため，グラフサイズが大きくなり，長時間のシーケンスに対して処理することが困難であり，計算コストが増加するといった問題がある．そのため，Graph Cuts を用いたビデオセグメンテーション手法では，これらの問題を解決するため，前処理として各ピクセルに対してクラスタリングを行い，各クラスタの代表ベクトルをノードとしてグラフを作成する [6] [7]．

本研究では永橋らの手法した平滑化処理の繰り返しによるグラフカットを動画像へ拡張するため，Spatio-temporal volume を用いた繰り返し処理によるグラフカットのビデオセグメンテーションを提案する．本手法では，入力された動画像の時空間データに対して Mean Shift クラスタリングを行い Spatio-temporal volume を作成する．Mean Shift クラスタリングのバンド幅を変化させて作成した Spatio-temporal volume に対し，バンド幅が大きなものから Graph Cuts によるセグメンテーションを行う．セグメンテーション結果から空間的確率と色分布確率を計算し，それを次の Graph Cuts の t-link に利用することで，大域的な情報を局所的な情報へ伝播させる．これらを繰り返し処理することにより，大域的なセグメンテーションから段階的に局所的なセグメンテーションを実現する．

2. Graph Cuts によるビデオセグメンテーション

本章では，ビデオセグメンテーションの従来法である Interactive Graph Cut [2] [3] について説明する．

2.1 Graph Cuts

動画像 P に対する各ピクセルを $p \in P$ としたとき，ラベルを $\boldsymbol{L} = \{L_1, L_2, \ldots, L_p, \ldots, L_{|P|}\}$ とし，各 L_p には物体 (“obj”) か背景 (“bkg”) のラベルが与えられる．また，p の近傍ピクセルを $q \in N$ とする．Graph Cuts では，エネルギー関数を式 (1) のように定義する．

$$E(\boldsymbol{L}) = \lambda \cdot R(\boldsymbol{L}) + B(\boldsymbol{L}) \tag{1}$$

λ は，$R(\boldsymbol{L})$ と $B(\boldsymbol{L})$ の比率のパラメータである．$R(\boldsymbol{L})$ は領域に対するペナルティ関数，$B(\boldsymbol{L})$ は物体と背景の境界に対するペナルティ関数であり以下に示すように定義する．

$$R(\boldsymbol{L}) = \sum_{p \in P} R_p(L_p) \tag{2}$$

$$B(\boldsymbol{L}) = \sum_{\{p,q\} \in N} B_{\{p,q\}} \cdot \delta(L_p, L_q) \tag{3}$$

$$\delta(L_p, L_q) = \begin{cases} 1 & \text{if} \quad L_p \neq L_q \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

$R_p(L_p)$ は，ピクセル p がラベル L_p である確率が高ければ値が小さくなるような関数として定義する．$B_{\{p,q\}}$ は，p と q の輝度値が似ていれば大きな値を出力する関数として定義する．$R(L)$ と $B(L)$ により定義したエネルギー関数 $E(\boldsymbol{L})$ を最小とするようなラベル $\boldsymbol{L}$ を Graph Cuts Algorithm [1] を用いて計算することでセグメンテーションを行う．

Graph Cuts Algorithm では，動画像から図 1 のようにグラフを作成し，min-cut/max-flow algorithm [1] を用いてグラフを分割する．グラフ G は，動画像の各ピクセルに対応したノードと source と sink と呼ばれるターミナルからなる．各ノード間を接続するエッジを n-link と呼び，各ノードから source (S), sink(T) のターミナルに接続するエッジを t-link と呼ぶ．n-link と t-link のエッジコストは，表 1 に従い設定する．

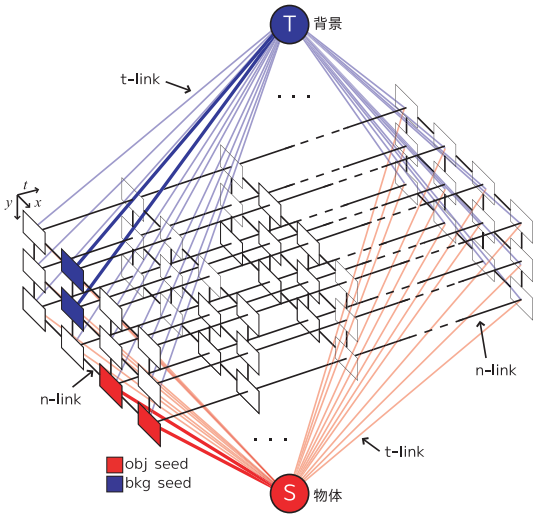


図 1　従来法 [3] による動画像のグラフ表現

表 1　エッジに与える重み

edge		cost	for
n-link	$\{p,q\}$	$B_{\{p,q\}}$	$\{p,q\} \in N$
t-link	$\{p,S\}$	$\lambda \cdot R_p(\text{"}bkg\text{"})$	$p \in P, p \notin \mathcal{O} \cup \mathcal{B}$
		K	$p \in \mathcal{O}$
		0	$p \in \mathcal{B}$
	$\{p,T\}$	$\lambda \cdot R_p(\text{"}obj\text{"})$	$p \in P, p \notin \mathcal{O} \cup \mathcal{B}$
		0	$p \in \mathcal{O}$
		K	$p \in \mathcal{B}$

このとき，

$$R_p(\text{“}obj\text{”}) = -\ln \Pr(I_p|\mathcal{O}) \tag{5}$$

$$R_p(\text{“}bkg\text{”}) = -\ln \Pr(I_p|\mathcal{B}) \tag{6}$$

$$B_{\{p,q\}} \propto \exp\left(-\frac{(I_p - I_q)^2}{2\sigma^2}\right) \frac{1}{dist(p,q)} \tag{7}$$

$$K = 1 + \max_{p \in P} \sum_{q:\{p,q\} \in N} B_{\{p,q\}} \tag{8}$$

となる．$\mathcal{O}$ は物体, $\mathcal{B}$ は背景を意味し, I_p はピクセル p の輝度値である．ユーザは一部のピクセルに対して $\mathcal{O}, \mathcal{B}$ を入力する．このとき入力された $\mathcal{O}, \mathcal{B}$ を seed と呼ぶ．$\Pr(I_p|\mathcal{O}), \Pr(I_p|\mathcal{B})$ は seed 以外のピクセルの t-link に設定する物体と背景の尤度である．$dist(p,q)$ はピクセル p, q のユークリッド距離を用いる．作成したグラフに対して，min-cut/max-flow algorithm を用いることで，物体と背景にグラフを分割する．このような処理により，ビデオセグメンテーションを実現する．

2.2 従来法の問題点

Interactive Graph Cuts [3] によるビデオセグメンテーションでは，各ピクセルをノードとして，グリッド上に n-link を作成する．t-link は，動画像の全てのピクセルに対して作成する．このように動画像に対してグラフを作成した場合，各ピクセルをノードとするため，グラフのサイズが大きくなる．例えば，解像度が 360×240, フレーム数が 100 とした時，作成されるエッジは約 2,500 万本となる．そのため，長時間のシーケンスに対応するには多くのメモリが必要となり，計算コストの増加などの問題が生じる．通常，このような問題を解決する手法として，動画像を時空間ボリュームデータとしてクラスタリングして作成した Spatio-temporal volume の各ボリュームをノードとすることでグラフのサイズを削減している．Spatio-temporal volume を用いた場合，各ボリューム単位でのセグメンテーションしか行えないため，高精度なセグメンテーションは期待できない．そこで本手法では，永橋らによって提案された平滑化の繰り返し処理によるグラフカットによるセグメンテーション [5] を，Spatio-temporal volume に拡張することで，大域的なセグメンテーションから局所的なセグメンテーションを実現する．

3. Spatio-temporal volume を用いた繰り返し処理によるグラフカット

Spatio-temporal volume とは，動画像を時系列のボリュームデータとして属性が似ているピクセルをクラスタリングしたものである．本手法では，異なるバンド幅で作成された Spatio-temporal volume を使用し，バンド幅が大きなものから Graph Cuts を繰り返し行うことにより，大域的なセグメンテーションから局所的なセグメンテーションを実現する（図 2）．

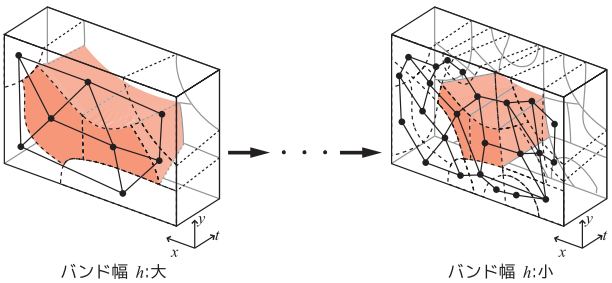


図 2　Spatio-temporal volume を用いた大域的なセグメンテーションから局所的なセグメンテーション

3.1 手法の流れ

図 3 に，提案手法の流れを示す．はじめに，入力された動画像に対してユーザが物体と背景の seed の入力を行う．次に，入力された動画像に対して，バンド幅 h で Mean Shift クラスタリングを行い，Spatio-temporal volume を作成する．作成された Spatio-temporal volume をノードとして Graph Cuts によりセグメンテーションを行う．セグメンテーション結果から，物体領域と背景領域の色分布に対して GMM(Gaussian Mixture Model) の当てはめ，色分布確率を計算する．また，物体領域と背景領域に対して距離変換を行い各ピクセルごとに物体と背景の空間的確率を計算する．色分布確率と空間的確率から事後確率を計算し，それをグラフの t-link のエッジコストを設定し，次の Graph Cuts 処理に利用する．バンド幅 h を $h = \alpha \cdot h$ として更新し，h が一定値以下になるまで繰り返し処理する．

これらの手順を以下に示す．

Step1. seed の入力

Step2. Spatio-temporal volume の作成

Step3. 対応点探索

Step4. Graph Cuts によるセグメンテーション

Step5. セグメンテーション結果から色分布確率・空間的確率の計算

Step6. $h = \alpha \cdot h (0 < \alpha < 1)$ と更新し h が一定値以下なら終了．それ以外なら **Step 2 へ**

以下に，各処理の詳細を記述する．

3.2 Spatio-temporal volume からグラフの作成

Spatio-temporal volume を用いた場合，移動量が大きい物体はフレーム間で異なるボリュームに分けられる可能性があり，隣接するボリュームのみでは最適なグラフを作成することが困難となる．これらの問題を解決するため，本手法では，n-link として隣接するボリュームとフレーム間の対応点探索から得られるボリュームの 2 種類のエッジを設定する．提案する Graph Cuts のエネルギー関数は以下のように定義する．

$$E(\boldsymbol{L}) = \lambda \cdot R(\boldsymbol{L}) + B_N(\boldsymbol{L}) + B_C(\boldsymbol{L}) \tag{9}$$

$$R(\boldsymbol{L}) = \sum_{p \in P} R_p(L_p) \tag{10}$$

$$B_N(\boldsymbol{L}) = \sum_{\{p,q\} \in N} B_{\{p,q\}} \cdot \delta(L_p, L_q) \tag{11}$$

$$B_C(\boldsymbol{L}) = \sum_{\{p,q\} \in C} B_{\{p,q\}} \cdot \delta(L_p, L_q) \tag{12}$$

ここで，$p, q \in P$ は Mean Shift クラスタリングにより得られる各 Spatio-temporal volume，N は隣接関係のボリュームの対の集合である，C はフレーム間の対応点から得られるボリュームの対の集合である．$B_C(\boldsymbol{L})$ を加えることにより，同一領域がフレーム間で異なるボリュームがつけられる場合でも，頑健なセグメンテーションが可能となる．

3.2.1 Spatio-temporal volume の作成

Mean Shift クラスタリング [8] を時空間に対して行うことで，Spatio-temporal volume を作成する．空間，時間，色情報を持った入力ベクトルを $\mathbf{x}_i = \{\mathbf{x}_i^s, \mathbf{x}_i^t, \mathbf{x}_i^r\}$，フィルタ結果を $\mathbf{z}_i$，各ピクセルのラベルを L_i とする．また，$\{\mathbf{y}_j\}_{j=1,2,\ldots}$ を以下のように定義する．

$$\mathbf{y}_{j+1} = \frac{\sum_{i=1}^{n} \mathbf{x}_i g\left(\left\|\frac{\mathbf{y}_j - \mathbf{x}_i}{h}\right\|^2\right)}{\sum_{i=1}^{n} g\left(\left\|\frac{\mathbf{x} - \mathbf{x}_i}{h}\right\|^2\right)} \tag{13}$$

$$g(\mathbf{x}) = \frac{C}{h_s^2 h_t h_r^p} k\left(\left\|\frac{\mathbf{x}^s}{h_s}\right\|^2\right) k\left(\left\|\frac{\mathbf{x}^t}{h_t}\right\|^2\right) k\left(\left\|\frac{\mathbf{x}^r}{h_r}\right\|^2\right) \tag{14}$$

ここで，h_s, h_t, h_r はそれぞれ空間，時間，色のバンド幅で，C は正規化定数である．$k(\mathbf{x})$ はカーネル関数である．Mean Shift クラスタリング は以下の手順で行われる．

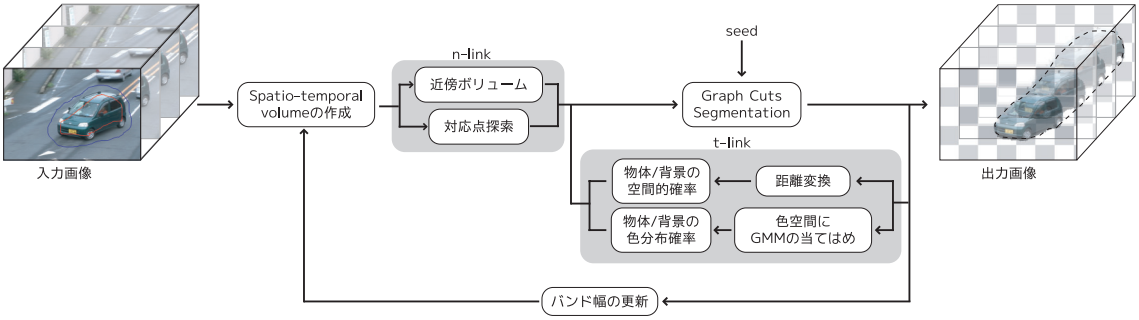


図 3　手法の流れ

（1）　$j = 0$ とし $y_{i,j} = x_i$ として初期化

（2）　$y_{i,j+1}$ を式 (13) により計算し，収束する $y_{i,c}$ まで繰り返し，$\mathbf{z}_i = \mathbf{y}_{i,c}$ として特徴空間の情報を格納

（3）　特徴空間において，$\mathbf{z}_i$ をクラスタ $\{C_p\}_{p=1,\ldots,m}$ にクラスタリング

（4）　$L_i = \{p | \mathbf{z}_i \in C_p\}$

（5）　空間領域において，M pixel 以下の領域を削除

この Mean Shift クラスタリングのバンド幅 h_s と h_t を大きいものから処理することにより，大域的なセグメンテーションから段階的に局所的なセグメンテーションを行う．図 4 に，各バンド幅における Spatio-temporal volume の例を示す．図 4

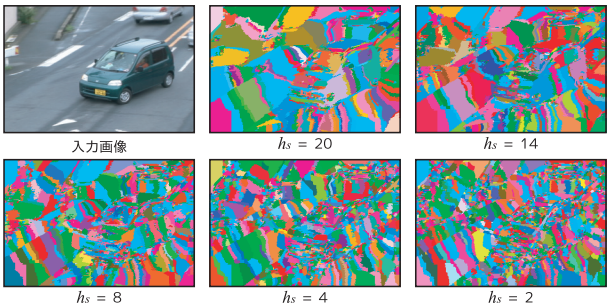


図 4　Spatio-temporal volume の作成例

から，バンド幅を大きな値から小さくすることで，各ボリュームが徐々に細かくなっているのがわかる．本手法では，このようにバンド幅を変化させることで大域的な情報から局所的な情報を表現する．Mean Shift クラスタリングにより作成された各ボリュームをノードとし，隣接するボリューム間にエッジを設定することで，グラフを作成する．

3.2.2　対応点探索によるエッジの追加

Mean Shift クラスタリングで Spatio-temporal volume を作成した際に，移動量が大きい物体はフレーム間で異なるボリュームに分けられる可能性がある．図 5 は，作成された Spatio-temporal volume を表しており，各ボリュームを色で表現している．図 5(a) のように物体の移動量が大きい場合，図 5(b) の時系列方向に見たときに同一物体のボリュームが隣接しない．そのため，隣接するボリュームのみでは最適なグラフを作成することが困難となる．本手法では，この問題を解決するために，フレーム間で対応点探索を行い，離れたボリューム間のエッジを設定する．対応点は，SIFT [9] により特徴点の検出をし，各点の SIFT 特徴量の類似度により求める．図 5(a) に示すように，マッチングにより得られる対応点を含む隣接するボリューム間にエッジを追加する．t フレームから $t+1$ フレームで領域が大きく移動した場合，隣接したボリュームだけでは同じ領域を繋ぐエッジは作成されないが，対応点探索からボリュームの対応を得ることで，エッジを作成し，フローを流すことが可能となる．

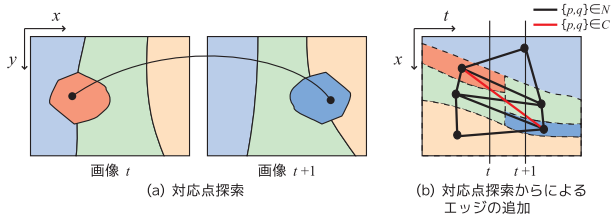


図 5　対応点探索によるエッジの追加例

3.3　繰り返し処理による Graph Cuts

Mean Shift クラスタリングのバンド幅を変化させた Spatio-temporal volume に対し，バンド幅が大きなものから Graph Cuts によるセグメンテーションを行う．セグメンテーション結果から空間的確率と色分布確率を計算し，それを Graph Cuts の t-link に利用することで，大域的な情報を局所的な情報へ伝播させる．以下に，n-link と t-link の更新方法と繰り返し処理の効果について述べる．

3.3.1　n-link の更新

n-link は，ノード間の関係を表す項である．本手法では，n-link に 3.2.1 により作成した隣接するボリューム間を表す $B_N(\boldsymbol{L})$ と，3.2.2 により追加した対応点探索により得られるボリューム間を表す $B_C(\boldsymbol{L})$ の 2 つを用いる．$B_N(\boldsymbol{L})$, $B_C(\boldsymbol{L})$ のそれぞれのエッジコストはボリューム間の色の類似度から計算する．ボリューム間の p, q のエッジコスト $B_{\{p,q\}}$ は以下の式により求める．

$$B_{\{p,q\}} = \exp\left(-\frac{\|\boldsymbol{I}_p - \boldsymbol{I}_q\|^2}{2\sigma^2}\right) \tag{15}$$

ここで，$\boldsymbol{I}$ は Spatio-temporal volume の代表色であり，Mean Shift クラスタリングで求められるクラスタの代表ベクトルにより決定する．本手法では，隣接するボリューム $B_N(\boldsymbol{L})$ と，対応点探索により得られるボリューム $B_C(\boldsymbol{L})$ を近傍ノードとしてグラフを作成する．

3.3.2　t-link の更新

t-link は，各ボリュームに対応するノードがどちらのラベルが割り当てられるかを表す項である．従来の Graph Cuts では，t-link は式 (5), (6) の尤度を用いて計算する．これを本手法では 1 回前の Graph Cuts の結果から事後確率を以下の式により

計算し，t-link に用いる．

$$R'_p(\text{“}obj\text{”}) = -\ln \Pr(\mathcal{O}|\boldsymbol{I}_p) \tag{16}$$

$$R'_p(\text{“}bkg\text{”}) = -\ln \Pr(\mathcal{B}|\boldsymbol{I}_p) \tag{17}$$

このとき，$\Pr(\mathcal{O}|I_p)$ と $\Pr(\mathcal{B}|I_p)$ はベイズの定理から式 (18), (19) により計算する．

$$\Pr(\mathcal{O}|\boldsymbol{I}_p) = \frac{\Pr(\mathcal{O})\Pr(\boldsymbol{I}_p|\mathcal{O})}{\Pr(\boldsymbol{I}_p)} \tag{18}$$

$$\Pr(\mathcal{B}|\boldsymbol{I}_p) = \frac{\Pr(\mathcal{B})\Pr(\boldsymbol{I}_p|\mathcal{B})}{\Pr(\boldsymbol{I}_p)} \tag{19}$$

分母の $\Pr(\boldsymbol{I}_p)$ は $\Pr(\mathcal{O})$ と $\Pr(\mathcal{B})$ の両方にあるため，これらの大小関係には影響がないので無視することができる．尤度 $\Pr(\boldsymbol{I}_p|\mathcal{O}), \Pr(\boldsymbol{I}_p|\mathcal{B})$ は従来法と同様に色分布確率を用いる．事前確率 $\Pr(\mathcal{O}), \Pr(\mathcal{B})$ には，1 つ前のセグメンテーション結果から空間的確率を計算する．空間的確率を用いることで，1 つ前のセグメンテーション結果に基づいて，次のセグメンテーションの注目位置を変化させる．この色分布確率と空間的確率を繰り返し処理の中で更新し，t-link に反映させることで，大域的なセグメンテーションから局所的なセグメンテーションを行う．

a） 色分布確率の更新

色分布確率 $\Pr(\boldsymbol{I}_p|\mathcal{O}), \Pr(\boldsymbol{I}_p|\mathcal{B})$ の計算には，GMM (Gaussian Mixture Model) [10] を用いて色分布を表現する．当てはめる色分布は L*a*b*の 3 次元となるため，以下の式を用いる．

$$\Pr(\boldsymbol{I}_p|\cdot) = \sum_{i=1}^{K} \alpha_i p_i(\boldsymbol{I}_p|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \tag{20}$$

$$p(\boldsymbol{I}_p|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{3/2}|\Sigma|^{1/2}} \cdot \exp\left(\frac{1}{2}(\boldsymbol{I}_p - \boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{I}_p - \boldsymbol{\mu})\right) \tag{21}$$

GMM を物体と背景の各領域の色分布に対して EM アルゴリズム [11] を用いて当てはめる．これにより，ヒストグラムでは離散的であった値に対して，GMM を用いることにより連続的に取り扱うことができる．当てはめた GMM のパラメータから式 (20) を用いることにより，$\boldsymbol{I}_p$ の色分布確率 $\Pr(\boldsymbol{I}_p|\mathcal{O})$ と $\Pr(\boldsymbol{I}_p|\mathcal{B})$ を求める．

b） 空間的確率の更新

1 つ前の Graph Cuts のセグメンテーション結果の空間情報を用いて，空間的確率 $\Pr(\mathcal{O}), \Pr(\mathcal{B})$ の更新を行う．物体と背景の境界付近では，次のセグメンテーション時にはどちらになるか不明なため，同程度に物体と背景が生起すると考えられる．一方，境界から離れている位置では，次のセグメンテーション時に物体と背景が入れ変わる可能性は低い．このような確率分布を作成するために，物体領域と背景領域に対して距離変換を行う．時空間の 3 次元に対してピクセル単位で，3 次元距離変換 [12] を行う．このとき，境界からの距離 d を 0.5 から 1 に正規化を行う．正規化をした物体の距離 d_{obj} と背景の距離 d_{bkg} を用いて空間的確率を以下の式で更新する．

$$\Pr(\mathcal{O}) = \begin{cases} d_{obj} & \text{if} \quad d_{obj} \geq d_{bkg} \\ 1 - d_{bkg} & \text{if} \quad d_{obj} < d_{bkg} \end{cases} \tag{22}$$

$$\Pr(\mathcal{B}) = 1 - \Pr(\mathcal{O}) \tag{23}$$

各ピクセルに対して $\Pr(\mathcal{O}), \Pr(\mathcal{B})$ を計算し，Mean Shift クラスタリングにより得られた各 Spatio-temporal volume 内の平均値をその位置の空間的確率とする．

3.3.3 繰り返し処理

GMM により得られた色分布確率 $\Pr(\boldsymbol{I}_p|\mathcal{O}), \Pr(\boldsymbol{I}_p|\mathcal{B})$ と距離変換により得られた空間的確率 $\Pr(\mathcal{O}), \Pr(\mathcal{B})$ を用いて，式 (16) により物体の事後確率 $\Pr(\mathcal{O}|\boldsymbol{I}_p)$，式 (17) により背景の事後確率 $\Pr(\mathcal{B}|\boldsymbol{I}_p)$ を計算する．

物体の事後確率 $\Pr(\mathcal{O}|\boldsymbol{I}_p)$ から式 (16) により $\{p, T\}$ 間の t-link のコストを，背景の事後確率 $\Pr(\mathcal{B}|\boldsymbol{I}_p)$ から式 (17) により $\{p, S\}$ 間の t-link のコストをそれぞれ計算し，Graph Cuts 処理に利用する．これにより，1 つ前のセグメンテーション結果を次の Graph Cuts へ伝播させることができる．図 6 に，繰り返し処理によるセグメンテーション結果を示す．

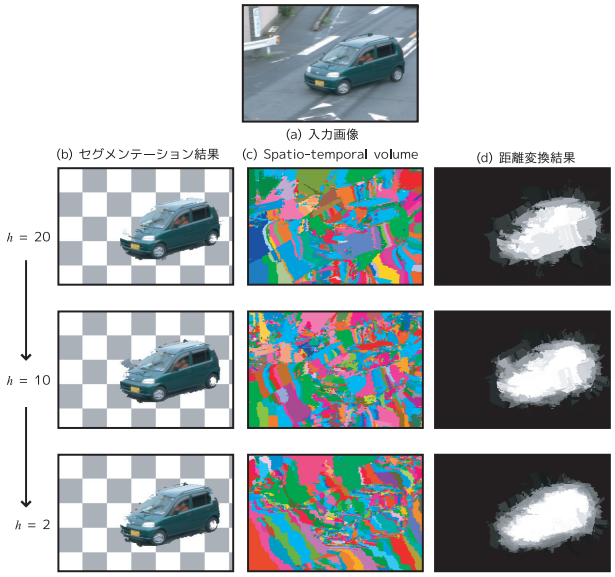


図 6 繰り返し処理によるセグメンテーション結果

4. 評価実験

4.1 実験概要

本手法の有効性を示すために評価実験を行う．非固定カメラで撮影し，撮影対象は徐々に見えが変化する動画像 3 つに対して実験を行う．入力する seed は初期フレームのみに与える．10 フレーム目のセグメンテーション結果から評価を行う．比較対象として，以下の手法と比較を行う．

従来法 1 各ピクセルをノードとしてグラフを作成する手法 [3]

従来法 2 Spatio-temporal volume をノードとしてグラフを作成する手法

提案手法 バンド幅を変化させた Spatio-temporal volume の繰り返し処理による手法

比較方法として再現率 (Recall) および適合率 (Precision) により評価する．物体抽出結果から，検出結果で正解した物体領域と背景領域の面積をそれぞれ True Positive(TP)，True Negative(TN) とする．また，背景領域を物体領域とした誤検

出の面積を False Positive(FP)，物体領域を背景領域とした誤検出の面積を False Negative(FN) とする．Recall, Precision, F-measure を以下の式により求める．

$$\text{Recall} = \frac{TP}{TP+FN} \tag{24}$$

$$\text{Precision} = \frac{TP}{TP+FP} \tag{25}$$

$$\text{F}-\text{measure} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \tag{26}$$

4.2 実験 1:従来法との比較

表 2 にセグメンテーション率，図 7 にセグメンテーション結果を示す． 表 2 より，従来法 2 のように Spatio-temporal volume を用いた場合は，ノード数が減少するため，1 ノードあたりの誤検出率が高くなる．そのため，従来法 1 の各ピクセルをノードとしたときよりセグメンテーション精度は低下する．提案手法は，Spatio-temporal volume のバンド幅を変化させて繰り返し処理することにより，従来法 2 と比較し約 0.10 セグメンテーション精度を向上させることができた．これは，従来法 1 と同程度のセグメンテーション精度であるため，セグメンテーション精度を低下させずに，従来法 1 よりグラフサイズを軽減させることができたといえる．表 3 に，画像サイズ 360×240，10 フレームの動画象に対してグラフを作成した際の各手法のノード数とエッジ数を示す．提案手法は，バンド幅 h を 20 から 2 まで変化させたときの結果である．従来法に比べ，提案手法はノード数を約 39 %，エッジ数を約 97 % 削減することがすることができた．

表 2 セグメンテーション率

	従来法 1 [3]	従来法 2	提案手法
Recall	0.88	0.82	0.92
Precision	0.96	0.77	0.88
F-measure	0.92	0.77	0.90

表 3 グラフのサイズ

	従来法 1 [3]	従来法 2	提案手法
ノード	864,000	52,993	43,140 ∼ 52,993
エッジ	2,505,600	81,239	17,477 ∼ 81,549

4.3 実験 2:対応点探索によるエッジの追加の効果

n-link に対応点探索によるエッジを追加した場合の効果を評価する．フレーム間の移動量が大きい場合，従来法では正確なセグメンテーションが困難となる．そこで，フレームレートが 6fps の動画像を用いて，対応点探索によるエッジを追加する手法と追加しない手法との比較実験を行う．表 4 に，実験結果を示す．実験結果より，対応点探索を用いることにより，大幅にセグメンテーション精度を改善することができた．これは，フレームレートが低いため，1 フレームでの移動量が大きくなるため，Mean Shift クラスタリングの際にフレーム間で同一領域が異なるクラスタに分けられるためである．そのため，隣接ボリュームのみで n-link を記述した場合，このようなケースに対応することができない．しかし，対応点探索を用いることで，離れたボリューム間の対応を得ることができるため，誤検出を抑制することができたと考えられる．

表 4 セグメンテーション率

	対応点探索なし	対応点探索あり
Recall	0.80	0.72
Precision	0.35	0.90
F-measure	0.49	0.80

5. ま と め

動画像のセグメンテーションでは，各ピクセルをノードとしてグラフを作成した場合，ノード数が増加するため長時間のシーケンスに対応できない，計算コストが増加するなどの問題がある．そこで，本稿ではバンド幅を変化させた Spatio-temporal volume を繰り返し処理することで，大域的なセグメンテーションから局所的なセグメンテーション手法を提案した．

評価実験により，各ピクセルをノードとする手法と Spatio-temporal volume を用いたのみの場合と比較し，バンド幅を変化させて繰り返し処理をすることにより，約 0.10 セグメンテーション精度を向上させることができた．これは，各ピクセルをノードとする場合と同程度の精度であり，セグメンテーション精度を低下させずに，従来法よりグラフサイズを軽減させることができたといえる．また，SIFT を用いた対応点探索結果を n-link に反映させることで，移動量が大きい場合でも頑健にセグメンテーションを行うことができることを確認した．

文 献

[1] Yuri Boykov and Vladimir Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 26, No. 9, pp. 1124–1137, 2004.

[2] Yuri Boykov and Marie-Pierre Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. *ICCV2001*, Vol. 01, p. 105, 2001.

[3] Yuri Boykov and Gareth Funka-Lea. Graph cuts and efficient n-d image segmentation. *Int. J. Comput. Vision*, Vol. 70, No. 2, pp. 109–131, 2006.

[4] 石川博. グラフカット (チュートリアル). 情報処理学会研究報告. CVIM (コンピュータビジョンとイメージメディア), Vol. 1, No. 31, pp. 193–204, 2007.

[5] 永橋知行, 藤吉弘亘, 金出武雄. 平滑化処理の繰り返しによる画像セグメンテーションのためのグラフカット. 第 10 回画像の認識・理解シンポジウム (MIRU2007), pp. 241–248, 2007.

[6] Jue Wang, Pravin Bhat, R. Alex Colburn, Maneesh Agrawala, and Michael F. Cohen. Interactive video cutout. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Papers*, pp. 585–594, New York, NY, USA, 2005. ACM Press.

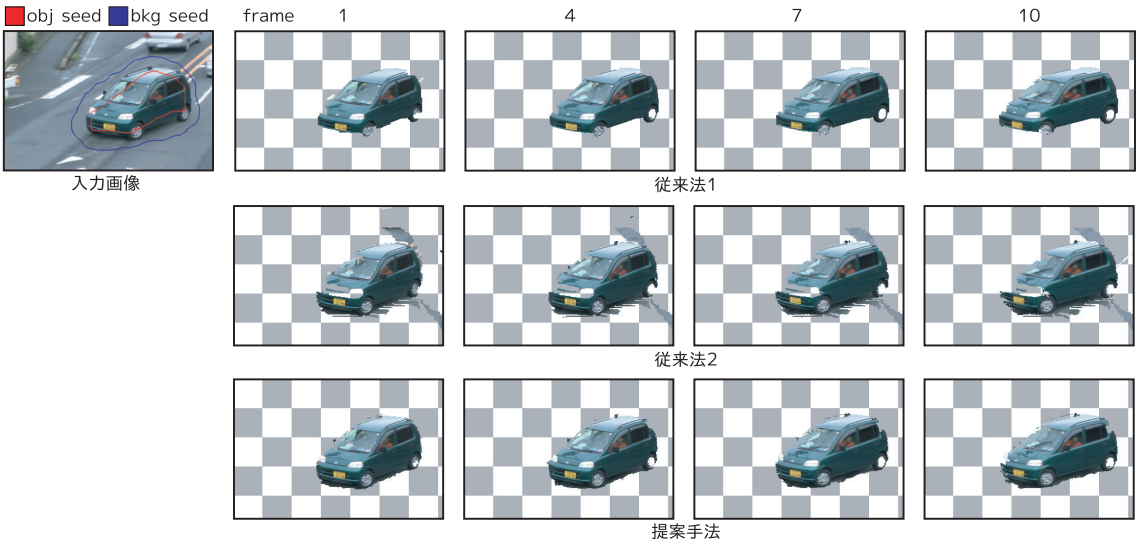
[7] Yin Li, Jian Sun, and Heung-Yeung Shum. Video object cut and paste. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Papers*, pp. 595–600, New York, NY, USA, 2005. ACM Press.

[8] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, No. 5, pp. 603–619, 2002.

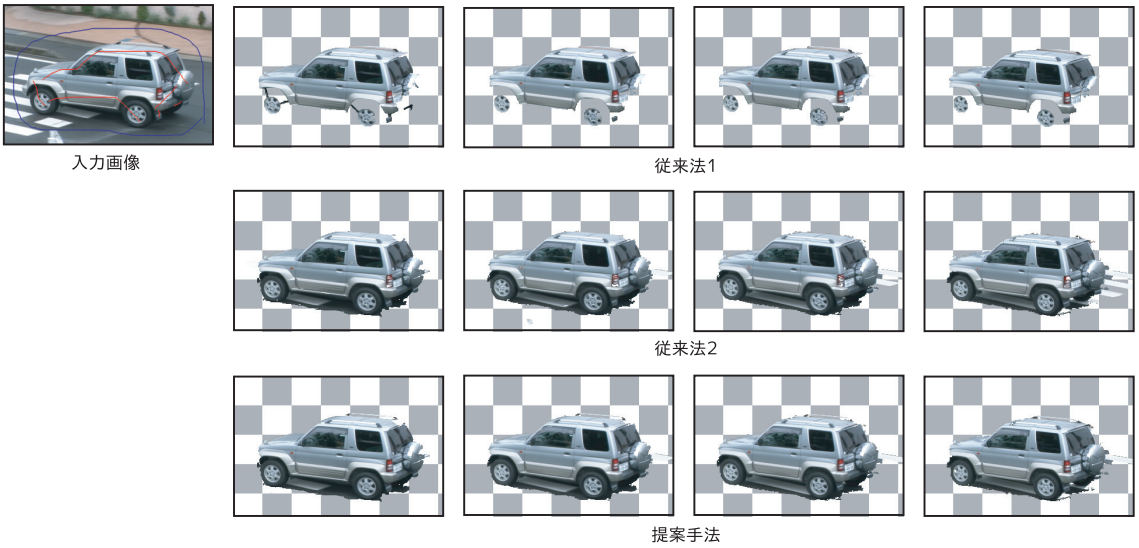
[9] David G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, Vol. 60, No. 2, pp. 91–110, 2004.

[10] C. Stauffer and W. E. L. Grimson. Adaptive background mixture models for real-time tracking. In *Proceedings of the IEEE Computer Science Conference on Computer Vision and Pattern Recognition (CVPR-99)*, pp. 246–252, Los Alamitos, June FebruaryMarch–FebruaryMay 1999. IEEE.

[11] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Jour-*



(a) シーケンス1



(b) シーケンス2

図 7　セグメンテーション結果

nal of the Royal Statistical Society. Series B (Methodological), Vol. 39, No. 1, pp. 1–38, 1977.

[12] 豊文斉藤, 純一郎鳥脇. 3 次元ディジタル画像に対するユークリッド距離変換. 電子情報通信学会論文誌. D-II, 情報・システム, II-情報処理, Vol. 76, No. 3, pp. 445–453, 19930325.