

## クロスエントロピーを用いた複数移動物体追跡の高速化

上 條 秀 一<sup>†1</sup> 宮 島 雄 一<sup>†1</sup> 松 井 淳<sup>‡2</sup>  
中 田 洋 平<sup>†1</sup> 松 本 隆<sup>†1</sup>

パーティクルフィルタを用いた複数移動物体追跡において、尤度関数にクロスエントロピーを組み込んだ追跡手法を提案し、高速化を図る。従来、尤度関数に Bhattacharyya 距離や KL 距離などを用いた手法が存在するが、これらは各時刻でのサンプルのヒストグラム生成に時間がかかる。これに対し、クロスエントロピーを組み込む手法では、サンプルのヒストグラムを生成せずに尤度計算を実行できるため、高速化が期待できる。Bhattacharyya 距離（従来手法）による手法との比較実験により提案手法の有効性、並びにサンプル数による影響を検証する。

### High-Speed Multiple Moving Objects Tracking by using Cross Entropy

SHUICHI KAMIJO,<sup>†1</sup> YUICHI MIYAJIMA,<sup>†1</sup>  
ATSUSHI MATSUI,<sup>‡2</sup> YOHEI NAKADA<sup>†1</sup>  
and TAKASHI MATSUMOTO<sup>†1</sup>

For multiple moving objects tracking using a particle filter, a tracking method in which cross entropy is incorporated into a likelihood function is proposed, with the aim of improving the tracking speed. Baseline methods have utilized Bhattacharyya distance, KL divergence, and so on, in the likelihood function. However, these methods require unnegligible computational cost in calculation of color histograms for each sample, drawn at each frame. In contrast, in the cross entropy method, likelihood calculations can be performed without generating sample histograms, which is expected to speed-up the tracking speed. The proposed method was experimentally compared with a baseline method using the Bhattacharyya distance. The effectiveness of the proposed method and the effect of the number of sample were examined.

### 1. はじめに

近年、コンピュータによる計算能力の向上や、ビデオカメラの急速な小型化、低コスト化に加え、応用範囲の広さから多種多様な業界で動画像追跡技術が盛んに研究されている。その応用範囲は、放送業界のスポーツ中継におけるボールや選手の自動追跡<sup>1)</sup> やセキュリティ分野における不審者検出<sup>2)</sup>、販売店などにおけるマーケティング<sup>3)</sup>、デジタルカメラにおける被写体追跡のソフトウェアなど大規模なシステムから、コンシューマ向け商品への実用化に至るまで非常に多岐に渡る。本研究では、動画像追跡において時々刻々と移動する追跡対象物が 2 つ以上の複数移動物体追跡について考える。

動画像追跡の追跡アルゴリズムには様々な手法が提案されているが、パーティクルフィルタ<sup>4)</sup> の有用性が近年数多く報告されている<sup>5)6)</sup>。しかし実用化に関しては処理時間の面で改善の余地がある。その主な要因のひとつとして、サンプルの総数に起因する処理コストが挙げられる。これまで提案されてきたパーティクルフィルタ動画像追跡アルゴリズムでは、しばしば尤度関数に Bhattacharyya 距離や KL 距離などが用いられているため、各時刻での各サンプルのヒストグラムを生成する必要があり、計算コストが比較的高い。これに対し、Cross Entropy を組み込む手法では、サンプルのヒストグラムを生成せずに尤度計算を実行できるため、計算時間の改善が期待できる。

### 2. 逐次モンテカルロ動画像追跡（パーティクルフィルタ）

まず、この節では特徴量に RGB カラーヒストグラム、尤度計算に Bhattacharyya 距離を用いて定式化を行い、次節で尤度計算に Cross Entropy を用いる。定式化に際して本研究では追跡対象物が 2 つ以上の動画像追跡（複数移動物体追跡）を扱うため、以下では追跡対象物の総数を  $TN^{all}$  として記述する。

#### 2.1 複数追跡対象物の表現

時刻  $t$  における追跡番号  $tn$  の追跡対象物の矩形領域  $X_{tn}(t)$ 、すべての追跡対象物の矩形領域をセットにした  $X(t)$  をそれぞれ順に、式 (1)、式 (2) のように定義し、その具体例として追跡対象物が 2 つの場合を図 1 に示す。

<sup>†1</sup> 早稲田大学大学院 先進理工学研究科

Faculty of Advanced Science and Engineering, Waseda University

<sup>‡2</sup> NHK 放送技術研究所

NHK Science and Technology Research Laboratories

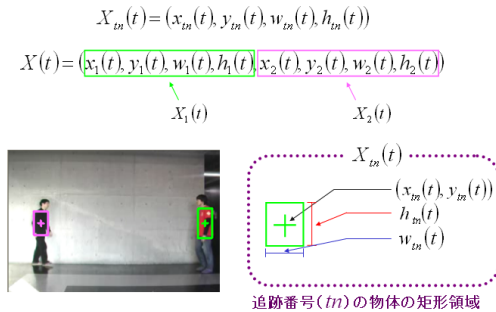


図 1 位置の定義  
Fig. 1 Definition of Position

$$X_{tn}(t) = (x_{tn}(t), y_{tn}(t), w_{tn}, h_{tn}) \quad (1)$$

$$X(t) = (X_1(t), X_2(t), \dots, X_{tn}(t), \dots, X_{TN^{all}-1}(t), X_{TN^{all}}(t)) \quad (2)$$

式 (1) において,  $x_{tn}(t), y_{tn}(t), w_{tn}, h_{tn}$  はそれぞれ追跡番号  $tn$  の追跡対象物の矩形領域の中心座標, 幅, 高さを表し, 追跡対象物や追跡候補の表現にこの矩形領域を用いる. ただし今回の実験で用いるテストデータの動画は全てカメラと追跡対象物との距離を固定して撮影を行ったため, 矩形領域の幅と高さである  $w$  と  $h$  は固定値としている.

次に時刻  $t$  における追跡番号  $tn$  の追跡対象物のサンプルの矩形領域  $X_{tn}^{(s)}(t)$ , 全物体のサンプルの矩形領域をセットにした  $X^{(s)}(t)$  をそれぞれ順に, 式 (3), 式 (4) のように表現する.

$$X_{tn}^{(s)}(t) = (x_{tn}^{(s)}(t), y_{tn}^{(s)}(t), w_{tn}^{(s)}, h_{tn}^{(s)}) \quad (3)$$

$$X^{(s)}(t) = (X_1^{(s)}(t), X_2^{(s)}(t), \dots, X_{tn}^{(s)}(t), \dots, X_{TN^{all}-1}^{(s)}(t), X_{TN^{all}}^{(s)}(t)) \quad (4)$$

式 (3), 式 (4) において,  $s = 1, \dots, N^{total}$ ,  $N^{total}$  はサンプルの総数である.

## 2.2 テンプレートの初期値計算

追跡番号  $tn$  の追跡対象物を表す矩形領域内のカラーヒストグラムの各 bin の値  $\hat{q}_{tn}(i)$  を次式のように, 各追跡物体ごとに保存する.

$$\hat{q}_{tn}(i) = \frac{\sum_{j \in R_{tn}} \delta_K(i, u(x_{tn,j}(0), y_{tn,j}(0)))}{n} \quad (5)$$

式 (5) 中の記号は以下の通りである.

$R_{tn} : (x_{tn}(0), y_{tn}(0))$  を中心とする矩形領域

$i$  : ヒストグラムの bin のラベル

$u$  : 画素の色情報を抽出する関数

$n$  : 矩形領域内の総 pixel 数

$\delta_K[x]$  : Kronecker のデルタ関数

(if,  $x=0$ , then,  $\delta_K[x] = 1$ , otherwise,  $\delta_K[x] = 0$ )

式 (5) によって計算されるカラーヒストグラムは R, G, B それぞれについて正規化されたカラーヒストグラムになる. 各物体の初期値計算は個々に行う. 本研究では, 初期値 ( $X_1(0), X_2(0)$ ) は手作業で与える.

初期サンプル  $X_{tn}^{(s)}(0) = (x_{tn}^{(s)}(0), y_{tn}^{(s)}(0), w_{tn}^{(s)}, h_{tn}^{(s)}) (s = 1, \dots, N^{total})$  は次のように生成する.

$$\begin{aligned} x_{tn}^{(s)}(0) &= x_{tn}(0) + \nu_{tn,x}, \nu_{tn,x} \sim N(0, 1) \\ y_{tn}^{(s)}(0) &= y_{tn}(0) + \nu_{tn,y}, \nu_{tn,y} \sim N(0, 1) \\ w_{tn}^{(s)} &= w_{tn} \\ h_{tn}^{(s)} &= h_{tn} \end{aligned} \quad (6)$$

式 (6) のようにサンプルの初期値は  $X_{tn}(0)$  の中心座標  $(x_{tn}(0), y_{tn}(0))$  に平均 0, 分散 1 の正規乱数を加えることにより生成する.

## 2.3 特徴量計算

式 (3) で生成したサンプル  $X_{tn}^{(s)}(t)$  の有する矩形領域内のカラーヒストグラム  $\hat{p}_{tn}^{(s)}(t, i)$  を各サンプルごとに計算する.

$$\hat{p}_{tn}^{(s)}(t, i) = \frac{\sum_{j \in R_{tn}^{(s)}(t)} \delta_K(i, u(x_{tn,j}^{(s)}(t), y_{tn,j}^{(s)}(t)))}{n} \quad (7)$$

式 (7) を計算する際に各フレームごとに生成されるヒストグラムの総数は, サンプルの総数  $N^{total}$ , 追跡対象物の数  $TN^{all}$  の積  $N^{total} \times TN^{all}$  である. なお, これらのヒストグラムは各サンプルの矩形領域内の総 pixel 数で正規化しているため, 正規化ヒストグラムとして計算される.

#### 2.4 Bhattacharyya 距離による尤度計算 (従来手法)

式 (5) で保存したテンプレートのヒストグラムと式 (7) で作成した各サンプルのカラーヒストグラムとの間の類似度を計算する事により, 各サンプルの尤度計算を行う. 本節では先行研究でしばしば用いられる Bhattacharyya 距離を用いる. 2 つの正規化された分布  $\hat{q}_{tn}$  と  $\hat{p}_{tn}^{(s)}(t)$  の間の Bhattacharyya 距離は式 (8) で与えられる.

$$Bhattacharyya(\hat{q}_{tn}, \hat{p}_{tn}^{(s)}(t)) = \sqrt{1 - \sum_{i=1}^m \sqrt{\hat{q}_{tn}(i) \times \hat{p}_{tn}^{(s)}(t, i)}} \quad (8)$$

これより, 各サンプルの尤度が定義される.

$$P(y_{tn}(t)|X_{tn}^{(s)}(t)) \propto \exp(-\lambda \times Bhattacharyya(\hat{q}_{tn}, \hat{p}_{tn}^{(s)}(t))) \quad (9)$$

式 (9) で求めた個々の物体の尤度を用い, 以下の式で全物体を統合した尤度を計算する.

$$P(y(t)|X^{(s)}(t)) = \prod_{tn=1}^{TN^{all}} P(y_{tn}(t)|X_{tn}^{(s)}(t)) \quad (10)$$

ここで,  $TN^{all}$  は追跡物体の数である. そして Importance Weight  $\Omega^{(s)}(t)$  を次式で定義する.

$$\Omega^{(s)}(t) = P(y(t)|X^{(s)}(t)) \quad (11)$$

最終的に, 追跡対象物の各時刻における位置の推定値は, 各サンプルが保持する矩形の座標値の期待値 (重み付け平均値) として次のように求められる. この期待値計算に際し, 式 (11) で与えられる Importance Weight を正規化した Normalized Importance Weight を重みとして用いる.

$$X(t) = \sum_{s=1}^{N^{total}} (X^{(s)}(t) \times \tilde{\Omega}^{(s)}(t)) \quad (12)$$

パーティクルフィルタによる追跡は以上の手順に従い, 毎フレームにおいて観測と予測を繰り返す事により追跡対象物の位置推定と次フレームにおける予測を行い各時刻における追跡を実現する.

### 3. Cross Entropy と背景ヒストグラムを用いた動画追跡

パーティクルフィルタを用いた動画追跡では十分な追跡の精度を得るためにはサンプル数を多く用意する必要がある. 前述の通りこの尤度計算を用いると各サンプルについてのヒストグラム生成 (式 (7)) とサンプルのヒストグラムとテンプレートのヒストグラム間の類似度計算 (式 (8)) を全てのサンプルについて行う必要があるため, 処理コストが大きい. 計算量のオーダーとしてはサンプルの総数  $N$  に対して Linearithmic -  $O(N \log N)$  である.

ここでは尤度関数に Cross Entropy を組み込んだ追跡手法を提案する. なお先行研究で追跡物体数が 1 物体の場合, 高速化が報告されている<sup>7)</sup>.

また頑健性を向上させる目的で背景ヒストグラムも尤度計算に考慮し, 背景も 1 つの特徴量として考える. 本研究では, 背景領域を「入力画像全領域から追跡対象物が少なくとも 1 物体存在する領域を除いた領域」と定義する. 背景も特徴とすることで追跡対象物の位置が背景領域となったサンプルに関して追跡対象物のみならず背景の尤度も低くなり, 実際の画像上の追跡対象物の位置が追跡対象物の領域となったサンプルの尤度がより高くなると考えられる. したがってオクルージョンが起こる複数移動物体追跡において, 背景ヒストグラムは有用な特徴量であると考えられる. なお, 本研究では追跡対象物間のオクルージョンのみを扱う. 追跡対象物の数は 1, 2, 3 とする.

#### 3.1 Cross Entropy に基づく尤度計算 (提案手法)

本研究では, 特徴量に RGB カラーヒストグラム, 尤度計算に Cross Entropy を用いる. 各サンプルの尤度関数は次式で与えられる.

$$P(y_{tn}(t)|X_{tn}^{(s)}(t)) \propto \exp(-\lambda \times H(\hat{p}_{tn}^{(s)}(t), \hat{q}_{tn})) \quad (13)$$

$$H(\hat{p}_{tn}^{(s)}(t), \hat{q}_{tn}) = - \sum_i \hat{p}_{tn}^{(s)}(t, i) \log(\hat{q}_{tn}(i)) \quad (14)$$

$$P(y(t)|X^{(s)}(t)) = \prod_{tn=1}^{TN^{all}} P(y_{tn}(t)|X_{tn}^{(s)}(t)) \times P(y_{back}(t)|X_{back}^{(s)}(t)) \quad (15)$$

ここで,  $H(\hat{p}, \hat{q})$  は 2 つの分布  $\hat{p}, \hat{q}$  の間の Cross Entropy を表し, 次式で与えられる.

$$\begin{aligned} H(\hat{p}, \hat{q}) &= - \sum_i \hat{p}(i) \log(\hat{q}(i)) \\ &= KL(\hat{p}, \hat{q}) + H(\hat{p}) \end{aligned} \quad (16)$$

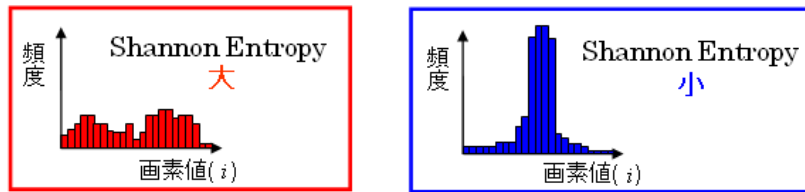


図 2 Shannon Entropy のイメージ  
Fig.2 Image of Shannon Entropy

式 (16) から Cross Entropy は KL 距離に加えて、サンプルのヒストグラム  $\hat{p}$  の Shannon Entropy  $H(\hat{p})$  の影響を考慮したもののみならずことができる。  $H(\hat{p})$  は  $\hat{p}$  の画素の種類が大きいほど大きくなり (乱雑な状態),  $\hat{p}$  の色の種類が少ないほど小さくなる (整理された状態) という特性を有している (図 2)。

つまり Cross Entropy を用いた尤度計算は次のような特徴があると考えられる。

- ・ 同じ KL 距離の  $\hat{p}$  の場合,  $\hat{p}$  の Shannon Entropy が小さい方が尤度が高い
- ・ 同じ Shannon Entropy 同士の  $\hat{p}$  の場合, KL 距離が小さい方が尤度が高い

それゆえ特性としては、追跡対象物の Shannon Entropy が小さく、背景の Shannon Entropy が大きい場合は追跡精度が良くなり、逆に追跡対象物の Shannon Entropy が大きく、背景の Shannon Entropy が小さい場合には追跡精度が悪くなると考えられる。

式 (13) を変形すると

$$P(y_{tn}(t)|X_{tn}^{(s)}(t)) \propto \exp(-\lambda \times - \sum_i \hat{p}_{tn}^{(s)}(t, i) \log(\hat{q}_{tn}(i))) \quad (17)$$

ここで提案手法の尤度計算はハイパーパラメータ  $\lambda$  を矩形領域内の総 pixel 数  $n$  で近似できると仮定して、両辺の対数を取って式変形すると

$$\begin{aligned} \log(P(y_{tn}(t)|X_{tn}^{(s)}(t))) &= \lambda \times (\sum_i \hat{p}_{tn}^{(s)}(t, i) \log(\hat{q}_{tn}(i))) \\ &= n (\sum_i \hat{p}_{tn}^{(s)}(t, i) \log(\hat{q}_{tn}(i))) \\ &= n \times \frac{1}{n} \times (\sum_x \log(\hat{q}_{bin}(x_i^{(s)}))) \quad (x : \text{pixel position}) \\ &= (\sum_x \log(\hat{q}_{bin}(x_i^{(s)}))) \end{aligned} \quad (18)$$

ここで、指数を取ると

$$P(y_{tn}(t)|X_{tn}^{(s)}(t)) \propto \prod_{i=1}^n (\hat{q}_{bin}(x_i^{(s)})) \quad (19)$$

したがって、この尤度計算はサンプルの矩形領域内の各 pixel が有する画素値のテンプレートのヒストグラムにおける頻度の積を計算している。

### 3.2 背景ヒストグラム

先行研究<sup>8)</sup>においてオクルージョン時の頑健性を向上させるため隠れ変数を用い、追跡対象物間の「手前」「奥」を考慮した手法が提案されている。手法を端的に述べると、人間の視覚系での情報処理能力を踏まえ、サンプルのヒストグラム生成の際に追跡対象物間の位置関係を考慮している (常に矩形領域すべての画素を用いるわけではない)。この手法で 2 物体追跡において、2 物体とも追跡精度の高精度化が確認されているが、この手法は追跡対象物が 2 物体の場合だから上手く働いた可能性がある。理由として隠れ変数の状態数が挙げられ、2 物体の場合は状態数が 3 つでよいものの、3 物体以上の場合には状態数の設定が複雑になると考えられる。このことから 3 物体以上においては状態数の設定に工夫が求められ、一般に追跡物体数が  $m$  つの場合でも上手く働く枠組みが求められる。そこで本研究ではオクルージョンを含め追跡時における頑健性の向上を図るため背景ヒストグラムを導入する。

本研究で用いるデータは比較的遠方から撮影していることもあり、追跡対象物の領域に対して背景領域の割合が多くなっている。そのような点から、追跡対象物同様のヒストグラムの生成方法を選択すると処理コストの面で問題が生じる。以下では、それを克服するために背景ヒストグラムの高速等価計算について述べる。

図 3 のように、追跡対象物が存在する領域を設定する。

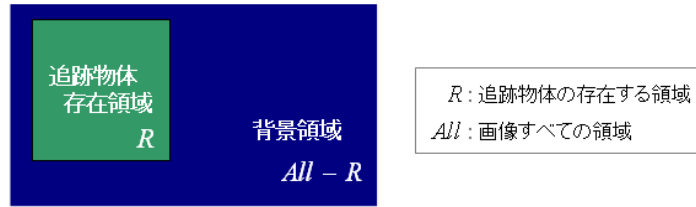


図 3 領域の定義  
Fig. 3 Definition of Area

尤度を  $L$  とすると、以下の式で計算できる。

$$L = \prod_{x \in R} h_1(c(x)) \times \prod_{x \notin R} h_2(c(x)) \quad (20)$$

ここで、 $(c(x))$  は色情報を抽出する関数である。次に、対数尤度を考え、式変形すると

$$\begin{aligned} \log L &= \sum_{x \in R} \log h_1(c(x)) + \sum_{x \notin R} \log h_2(c(x)) \\ &= \sum_{x \in R} \log h_1(c(x)) + \left\{ \sum_{x \in All} \log h_2(c(x)) - \sum_{x \in R} \log h_2(c(x)) \right\} \\ &= \sum_{x \in R} \{ \log h_1(c(x)) - \log h_2(c(x)) \} + const. \\ &= \sum_{x \in R} \log \frac{h_1}{h_2}(c(x)) + const. \end{aligned} \quad (21)$$

ここで、指数を取ると

$$L \propto \prod_{x \in R} \frac{h_1}{h_2}(c(x)) \quad (22)$$

この式は背景領域の色情報を取得せずともヒストグラムの生成方法を工夫すれば、追跡対象物領域内の色情報のみを取得することにより、背景情報も考慮した尤度を計算できることを示している。したがって、画像内の背景の大きさが追跡対象物の大きさと比較して大きい場合、本手法は大幅な処理コストの削減が期待できる。



1 フレーム      78 フレーム      134 フレーム

図 4 評価データ  
Fig. 4 Test Sequence

背景情報を考慮した各追跡対象物の尤度の式は Bhattacharyya 距離，Cross Entropy において順にそれぞれ次式のようになる。

$$P(y_{tn}(t)|X_{tn}^{(s)}(t)) \propto \prod_{j \in R_{tn}^{(s)}(t)} \exp \left\{ -\lambda \times \left( \sqrt{1 - \sum_{i=1}^m \sqrt{p_{tn}^{(s)}(t, i)} \times q_{tn}(i)} + \sqrt{1 - \sum_{i=1}^m \sqrt{p_{back}^{(s)}(t, i)} \times q_{back}(i)} \right) \right\} \quad (23)$$

$$P(y_{tn}(t)|X_{tn}^{(s)}(t)) \propto \prod_{j \in R_{tn}^{(s)}(t)} \frac{q_{tn}(t) \times \delta_K(i, u(x_{tn,j}^{(s)}(t), y_{tn,j}^{(s)}(t)))}{q_{back}(t) \times \delta_K(i, u(x_{back,j}^{(s)}(t), y_{back,j}^{(s)}(t)))} \quad (24)$$

今回は Cross Entropy の高速化に焦点を絞るため、動画像追跡に背景ヒストグラムがどのような影響を与えるのか調査した検証実験、並びに高速等価計算の詳細な記述は別紙に回す。なお以下の実験では従来手法（Bhattacharyya 距離）、提案手法（Cross Entropy）ともに背景ヒストグラムの高速等価計算を用いている。

#### 4. 実 験

従来手法の Bhattacharyya 距離による尤度計算（式 (23)）と、提案手法である Cross Entropy による尤度計算（式 (24)）を用いた場合の追跡の精度と処理コストを比較する。

実験で用いたデータを図 4 に示す。評価データの画像サイズは  $360 \times 240$ [pixel] でフレーム数は 180 フレームである。この評価データは青服の人物と赤服の人物、そして紫服の人物がそれぞれ自由に歩行する動画であり、追跡対象物は 3 物体ともに動体部分（トルソー）

とする．評価方法の累積誤差とは各フレームにおける誤差，これはあらかじめ人の手で画像をクリックする事により生成した正解値の座標と実際にアルゴリズムが推定した座標との画面上の直線距離の累積和 [pixel] である．

実験の条件を以下に示す．

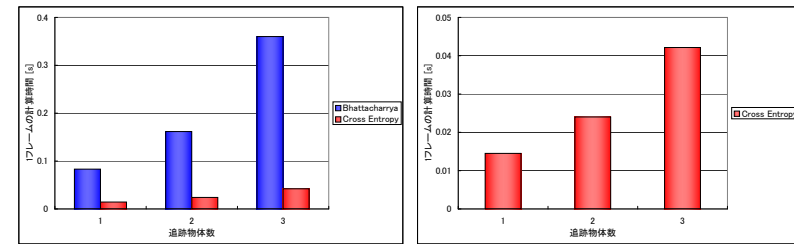
特徴量	RGB カラーヒストグラム (8 × 8 × 8)
評価尺度	従来手法 (式 (23)), 提案手法 (式 (24))
追跡アルゴリズム	パーティクルフィルタ
矩形領域の大きさ (全人物)	30 × 30[pixel]
定量評価	処理コスト [sec], 累積誤差 [pixel]
サンプル数	[100, 500]
追跡物体数	10 (従来手法のみ) [1, 2, 3]

実験環境は一般的な計算機であるノートパソコン (CPU : Intel Core(TM)2 CPU T5500(1.66[GHz]), メモリ : 0.99GB) を用いた．

まずサンプル数 100 の場合の計算時間を表 1 に表し，表 1 の関係を表すグラフを図 5 に示す．そしてサンプル数 100 の場合の累積誤差を表 2 に表す．次にサンプル数 500 の場合の計算時間を表 3 に表し，表 3 の関係を表すグラフを図 6 に示す．またサンプル数 500 の場合の累積誤差を表 4 に表す．

表 1 サンプル数 100 の計算時間  
 Table 1 Calculation Time (100 samples)

サンプル数	100					
	1		2		3	
評価尺度	Bh*3	Cr*4	Bh	Cr	Bh	Cr
計算時間 [sec]	15.0	2.61	29.1	4.32	64.8	7.59
1 フレームの計算時間 [spf]	0.0831	0.0145	0.162	0.024	0.36	0.042
Bh ÷ Cr[倍]	5.74		6.73		8.54	



従来手法 対 提案手法

提案手法

図 5 サンプル数 100 の計算時間

Fig. 5 Calculation Time (100 samples)

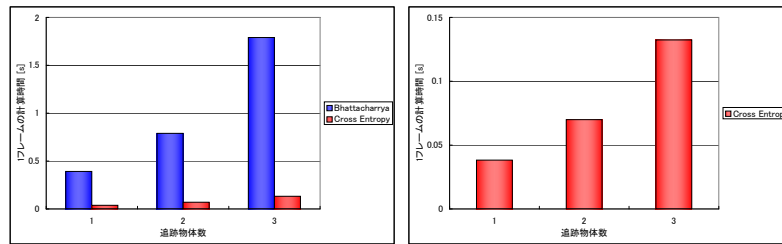
表 2 サンプル数 100 の累積誤差  
 Table 2 Cumulative Error (100 samples)

サンプル数	100					
	Bhattacharyya 距離			Cross Entropy		
追跡物体数	1	2	3	1	2	3
青服の累積誤差 [pixel]	748	760	732	554	581	621
赤服の累積誤差 [pixel]	-	878	837	-	344	354
紫服の累積誤差 [pixel]	-	-	904	-	-	546

\*3 Bhattacharyya 距離  
 \*4 Cross Entropy  
 \*5 Bhattacharyya 距離  
 \*6 Cross Entropy

表 3 サンプル数 500 の計算時間  
Table 3 Calculation Time (500 samples)

サンプル数	500					
	1		2		3	
追跡物体数	Bh*5	Cr*6	Bh	Cr	Bh	Cr
評価尺度	Bh*5	Cr*6	Bh	Cr	Bh	Cr
計算時間 [sec]	70.5	6.89	142	12.6	322	23.8
1 フレームの計算時間 [spf]	0.391	0.0383	0.789	0.07	1.79	0.132
Bh ÷ Cr[倍]	10.2		11.3		13.5	



従来手法 対 提案手法

提案手法

図 6 サンプル数 500 の計算時間

Fig. 6 Calculation Time (500 samples)

表 4 サンプル数 500 の累積誤差  
Table 4 Cumulative Error (500 samples)

サンプル数	500					
	Bhattacharyya 距離			Cross Entropy		
追跡物体数	1	2	3	1	2	3
青服の累積誤差 [pixel]	725	718	734	560	562	557
赤服の累積誤差 [pixel]	-	813	842	-	315	336
紫服の累積誤差 [pixel]	-	-	890	-	-	542

## 5. 考 察

まず両者の処理コストについて考察する．図 5 と図 6 を見ると，提案手法の尤度計算を用いた場合，処理コストが大幅に削減できていることがわかる．尤度に Bhattacharyya 距

離を用いた場合との追跡速度と比較すると，追跡物体数が 1, 2, 3 物体のいずれにおいてもサンプル数 100 の場合で 5 倍，サンプル数 500 の場合で 10 倍程度の高速度が確認できる．これは従来手法の尤度計算が各サンプルについて，ヒストグラムの生成（各ビンの変数用意，初期化など）を行い，そのヒストグラムを正規化し，重み付けした後にテンプレートのヒストグラムの各ビンの値 ( $8 \times 8 \times 8=512$ ) を全て参照し尤度の計算をおこなうのに対し，提案手法の尤度計算は予め用意したテンプレートのヒストグラムの値を参照するのみで尤度が求まるため大幅に処理コストを削減できたと考えられる．またサンプル数が多いほど，Cross Entropy による高速化の割合が高いこともわかる．その原因として，従来手法では計算機の性能が計算量の多さに追いついていない点が挙げられる．従来手法でサンプル数を 500 にした場合，1 フレーム目の処理から，CPU やメモリの占有率，キャッシュなどがほぼ 100 %となっていた．この点に関しては十分なメモリを積んだ計算機環境で更に検証を重ねる必要がある．また，各フレームの処理コストは主に，画像の入出力とアルゴリズムの処理コストから構成されるが図 5 と図 6 の処理コストは画像の入出力を含めた処理コストである．そのため実際には画像の入出力のみで 1 フレームあたり 0.011[sec] の処理コストがかかっており，この部分に関しては OpenCV<sup>9)</sup> などを用いることで高速化が可能であると考えられる．そのため実質的にはサンプル数 100 の場合において充分リアルタイムでの処理が可能な処理スピードといえる．

次に，追跡精度について考察する．表 2 と表 4 を見ると任意の追跡対象物について同サンプル数では，サンプル数が 100, 500 のいずれの場合も提案手法の追跡精度の方が良いことが分かる．これは今回のテストデータとして Cross Entropy の仮説に合致したもの（追跡対象物が単色，背景一定など）を選んだことが原因のひとつとして考えられる．逆に前述した Cross Entropy の特性上仮説に矛盾するデータ（対象物が多色など）を用いた場合は精度が低下する恐れがある．

## 6. ま と め

Cross Entropy の尤度計算を導入することにより，パーティクルフィルタの実装上の問題点のひとつであるサンプル数の尤度計算による処理コストへの負荷を軽減し，従来の Bhattacharyya 距離と比べて追跡物体数が 1, 2, 3 つのいずれの場合においてもサンプル数 100 の場合で 5 倍，サンプル数 500 の場合で 10 倍程度の高速度を得た．また Cross Entropy の追跡対象物の Shannon Entropy が小さい場合は追跡精度が良くなるという特性上，本データ（追跡対象物が単色）を用いた場合精度の向上も確認した．

### 参 考 文 献

- 1) 三須俊彦, 松井淳, 苗村昌秀: センサネットワークのための分散協調型パーティクルフィルタ: 多視点サッカーボール追跡への応用, Technical report of IEICE PRMU, vol.106, No.538 (2006).
- 2) 大澤達哉, 若林佳織: 映像モニタリングのための人物追跡技術, NTT 技術ジャーナル 2007.8 (2007).
- 3) 岡部亜梨子, 安倍満, 小沢慎治: AbS 手法を用いた店舗内における人物の検出”, 電気学会論文誌 C (電子・情報・システム部門誌), Vol.127 (2007).
- 4) Michael Isard et.al.: CONDENSATION-Condensation Density Propagation for Visual Tracking, International Journal of Computer Vision 29(1), 5-28 (1998).
- 5) 味八木崇, 山崎俊彦, 相澤清晴: Wi-Fi 位置推定と映像のパーティクルフィルタによる広域人物追跡, Technical report of IEICE Multimedia and virtual environment Vol.107, No.130 (2007).
- 6) 小林貴訓, 杉村大輔, 平澤宏祐: パーティクルフィルタとカスケード型識別器の統合による人物三次元追跡, 電子情報通信学会論文誌 D-II, Vol.J90-D-II, No.8, pp.1-11 (2007).
- 7) 宮島雄一, 松井淳, 中田洋平, 松本隆: ヒストグラムを用いたピクセル生成モデルにもとづく逐次モンテカルロ動画追跡, 信学技報, PRMU2007-105, pp.75-80 (2007).
- 8) Shuichi Kamijo, Yuichi Miyajima, Atsushi Matsui, Yohei Nakada, and Takashi Matsumoto: Hidden Occlusion Variable Approach to Multiple Object Tracking with Sequential Monte Carlo Implementation, Proc. of the 3rd Korea-Japan Joint Workshop on Pattern Recognition (KJPR2008), pp.69-70 (2008).
- 9) 奈良先端科学技術大学院大学: OpenCV プログラミング, 株式会社毎日コミュニケーションズ (2007).