

## 平均値シフト法を用いた複数物体追跡の研究

川中 大樹 中島 平 岡谷 貴之 出口 光一郎

本論文では画像中の複数物体をリアルタイムで追跡する手法を提案する。背景差分法を用いて対象物体を安定して画像から抽出できる場合には、得られた対象領域に対して平均値シフト法を用いることで、極めて少ない計算量ながらかなりロバストな対象の追跡が可能である。しかし、画像中に複数の移動物体が存在し、これらがたびたび交差するときには対象物体の区別が必要となる。そこで、オクルージョン発生時に対象領域中の色の分布に応じて確率的な重み付けを施した平均値シフト法を行うことで、対象物体を区別して追跡できるようにした。この重みはオクルージョンが起こったときに追跡対象の色のヒストグラムと重なった対象の色のヒストグラムと比較して作成する。ヒストグラムはオクルージョンが起こっていないときに求めて更新するので、予め対象人物の色情報を知る必要がない。

## On Tracking Multiple Objects Using Mean Shift Algorithm

Kawanaka Daiki, Taira Nakajima, Takayuki Okatani, Koichiro Deguchi

This paper presents a method for tracking multiple objects in an image sequence in a real time. If the objects can be accurately extracted from the images by background subtraction, it is possible to robustly track the objects with a small amount of computational complexity using the mean shift algorithm. When there are multiple moving objects in the images and they frequently overlap with each other, it is necessary to identify each of them correctly to track them. An algorithm for this problem is shown which is based on mean shift algorithm with probabilistic weights that are computed from color distribution of each target. The weights are generated at every time when occlusion occurs and they are computed from comparison between the histogram of the target window and those of the overlapped windows. The histograms are periodically updated when there is no occlusion.

### 1 はじめに

画像中の人物の位置や動作を知ることは、ビデオ監視システム等、様々な分野に応用できるものとして期待されている。動作とは対象の位置や形状等の変化であり、対象の動作を解析するにはそれらの変化が分からなければならない。そのため、人物を追跡する必要がある。

追跡対象が1人の場合には安定に追跡することが出来るが、追跡対象が複数の場合には対象同士がすれ違う時などにオクルージョンが起こる。オクルージョンが起こった場合には重なった追跡対象を区別して追跡する必要がある。そのような追跡方法として、抽出した輪郭を追跡する方法や固有空間のマッ

チングを利用する追跡方法等も提案されているが、これらの方法は一般に計算量が多く、リアルタイムでの処理にはあまり向いていない[2]。

リアルタイム性を重視する点から、少ない計算で比較的ロバストに追跡可能な平均値シフト法[1,2]が注目されている。文献[1]では、追跡すべきモデルの色の分布と画像中の領域の色の分布を比較して非剛体物体を追跡する方法が提案され、数学的に良い結果が得られている。しかしながら複数物体を追跡したときのオクルージョンの取り扱いが明示的に行われておらず問題が起る可能性がある。本研究ではオクルージョンを考慮した複数人物の追跡を行う方法として、色の分布を比較するのではなくオクルージョンが発生したときに色の分布を重みとして効果的に利用し、複数物体を区別して追跡する方法を提案する。本手法を用いて実画像中の複数人物の追跡を行い、オクルージョン時の追跡に対する有効性を検証した。

東北大学大学院情報科学研究科 宮城県仙台市青葉区荒巻字青葉

01

Graduated School of Information Sciences, Tohoku University, Aramaki-aza Aoba 01, Aoba-ku, Sendai, Miyagi

## 2 背景差分と平均値シフト法による追跡

### 2.1 背景差分による対象領域の抽出

追跡のための手がかりとして色やテクスチャといった情報を利用することも出来るが、本研究ではロバスト性を考慮して、まず第一に背景差分法を用いて対象領域を明示的に抽出し、追跡することを考える。背景差分法は照明変化など環境の変動には強くないが、室内など照明が整備された状況によっては強力な方法である。

背景差分法では抽出すべき対象が存在しない背景画像を予め用意しておく。そして入力画像と背景画像の各画素について輝度値の距離の差を計算し、この差が閾値より大きい領域を対象領域とする。本研究ではノイズ等の影響を減らすために背景画像として複数枚の背景画像の平均を用いた。また、閾値には複数枚の背景画像の画素  $(x, y)$  の輝度値の分散  $V(x, y)$  を用いて、画素ごとに閾値  $T(x, y)$  を決定した。

$$T(x, y) = \alpha \sqrt{V(x, y)} + \beta \quad (1)$$

ここで、 $\alpha, \beta$  は定数である。

### 2.2 平均値シフト法による対象の追跡

平均値シフト法はある関数  $f(x)$  の初期値周辺のある区間での傾きを求め、求めた傾きで  $f(x)$  の値が大きくなる方向へ区間の中心を移動させ、初期値周辺で  $f(x)$  が極大となる位置を求める方法である。本研究では追跡対象それぞれに追跡対象を囲むように長方形の枠(以下ウィンドウと呼ぶ)を配置し、毎フレーム追跡対象を囲むようにウィンドウを移動させることにより対象の追跡を行う。このウィンドウは画像の横軸(以下  $x$  軸)及び縦軸(以下  $y$  軸)に平行な長方形である。本研究では、関数は画素の点  $\mathbf{x}$  が対象か背景かを示す関数  $I(\mathbf{x})$  をあるウィンドウ関数で畳み込んだものとする。この関数に対して平均値シフト法を用いることで、ウィンドウ内で対象領域の多い方向、すなわち追跡対象が存在すると思われる方向にウィンドウは移動される。具体的には以下のような方法で追跡を行う。

1. 前フレームで得られたウィンドウの位置を現在のフレームでの初期位置とする。
2. ウィンドウ内で背景差分を行い対象領域の重心

$\mathbf{x}_g = (x_g, y_g)$  を求める。

$$\mathbf{x}_g = \frac{\sum_{i=1}^l \mathbf{x}_i \cdot I(\mathbf{x}_i)}{\sum_{i=1}^l I(\mathbf{x}_i)} \quad (2)$$

$$I(\mathbf{x}_i) = \begin{cases} 0 & (\text{画素 } \mathbf{x}_i \text{ が背景領域}) \\ 1 & (\text{画素 } \mathbf{x}_i \text{ が対象領域}) \end{cases} \quad (3)$$

ここでウィンドウ中には  $l$  画素存在するとし、 $\mathbf{x}_i$  はウィンドウ内の画素の位置である。

3. ウィンドウの中心を、求めた重心  $\mathbf{x}_g$  に移動させる。
4. ウィンドウの移動量が閾値  $T_h$  画素以下になるか、繰り返し回数がある回数以上になるまで手順 2, 3 を繰り返す

この方法ではウィンドウは最も近くの対象を追跡することになり、フレーム数が多く対象の移動量が小さければ、現在のフレームと前のフレームで同一の対象を追跡することが出来る。

ここで追跡対象の大きさは、前のフレームと現在のフレームでは異なることがありうる。ウィンドウの大きさが適切でない場合、対象の重心を正確に求められなかったり、他の対象を捉えやすくなったりする。そこで、適したウィンドウの大きさになるように、以下の式に基づいてウィンドウの大きさを修正する。

$$\mathbf{L} = a\mathbf{V} + b \quad (4)$$

ここで、 $\mathbf{L} = (L_x, L_y)$  は  $x$  軸、 $y$  軸方向のウィンドウの長さ、 $\mathbf{V} = (V_x, V_y)$  はウィンドウ中の対象領域の  $x$  軸、 $y$  軸方向の分散である。これにより求めたウィンドウを次のフレームに配置して同様の操作を繰り返し、対象を追跡する。

## 3 複数対象の追跡

単一の対象ならば前節の方法でうまく追跡することができる。次に  $n$  個の対象を追跡する場合を考える。ウィンドウ同士が重なると、あるウィンドウ中にそれが本来追跡する対象の他に、重なったウィンドウが追跡してきた対象も存在してしまう。ここでは、このような場合にもそれぞれの対象を区別しながら追跡できるように、対象領域中の色の分布を利用する。

### 3.1 対象の生起確率で重み付けを行う追跡

ウィンドウ同士が重なったとき、重み付きの平均値シフト法を行う。重みは背景差分で切り出された各画素に対して定義され、それはその画素が重なったウィンドウの中で今考えているウィンドウに属する確率に相当する。重なったウィンドウが追跡する対象での追跡対象である確率に基づくものとする。結果的に、平均値シフト法により追跡対象が存在する確率もっとも高い方向へ移動することになる。次節で述べるが、上述の確率は追跡対象の色の分布を元に推定したもので、重みはその画素の持つ色の関数となる。実際の処理方法として、ウィンドウ同士が重なった場合には以下の方法で追跡を行う。

1. 次節に述べる方法で重みを計算し、テーブルを作成する。
2. 前フレームで得られたウィンドウの位置を現在のフレームでの初期位置とし、ウィンドウの初期サイズの修正を行う。修正方法については3.4節で述べる。
3. ウィンドウ内で背景差分を行い、対象領域の場合にはその画素  $\mathbf{x} = (x, y)$  の輝度値から領域  $\mathbf{x}$  の色から領域  $\mathbf{x}$  に与える重み  $w(\mathbf{x})$  を決める。
4. 式 (5) により重みを考慮したウィンドウ中の対象領域  $\mathbf{x}_i$  の重心  $\mathbf{x}_g = (x_g, y_g)$  を求める。

$$\mathbf{x}_g = \frac{\sum_i \mathbf{x}_i^l \cdot w(\mathbf{x}_i) \cdot I(\mathbf{x}_i)}{\sum_{i=0}^l w(\mathbf{x}_i) \cdot I(\mathbf{x}_i)} \quad (5)$$

$$I(\mathbf{x}_i) = \begin{cases} 0 & (\text{画素 } \mathbf{x}_i \text{ が背景領域}) \\ 1 & (\text{画素 } \mathbf{x}_i \text{ が対象領域}) \end{cases}$$

次節で述べるが、 $w(\mathbf{x}_i) = w(q(\mathbf{x}))$  である。

5. ウィンドウの中心を求めた重心  $\mathbf{x}_g$  に移動させる。
6. ウィンドウの移動量が閾値  $T_h$  画素以下になるか、繰り返し回数がある回数以上になるまで手順4, 5を繰り返す

対象領域に与えた重みの和が0になる場合には対象が完全に重なったとみなして、重みの和が0となったウィンドウの中心を、重なったウィンドウの中心

に移動させる。また、この場合にはウィンドウの大きさを修正しないことにする。これは追跡対象のみの対象領域の分散を計算できないためである。

### 3.2 重みの計算方法

その色の分布を表現するのに離散的なヒストグラム表現を用いる。そのためHSV表色系のHS色空間を  $m$  個に分割する。この分割された色空間を色領域と呼ぶことにする。与える重みは対象領域の色から色領域と重みの関係を表したテーブルを参照することにより決定する。このテーブルはウィンドウ同士が重なったときに作成する。1つのウィンドウに  $p(p < n)$  個の異なるウィンドウが重なっており、注目するウィンドウは対象  $k_0$  を追跡中であり、他のウィンドウはそれぞれ  $k_1, \dots, k_p$  を追跡していたとする。重み  $w(\mathbf{x})$  として、その画素での色  $q(= 1, \dots, m)$  が、追跡対象  $k_0$  である確率  $\Pr(k_0|q)$  を用いるとする。これは、ベイズの定理より

$$\begin{aligned} \Pr(k_0|q) &= \frac{\Pr(k_0, q)}{\Pr(q)} \\ &= \frac{\Pr(q|k_0)\Pr(k_0)}{\sum_{i=0}^p \Pr(q|k_i)\Pr(k_i)} \quad (6) \end{aligned}$$

ここでは追跡対象の大きさの変化やオクルージョンによりある対象領域が追跡対象  $k_i$  である確率が分からないため、等確率であると仮定して、

$$\Pr(k_i) = \frac{1}{p+1} \quad (7)$$

とする。

ここで、 $\Pr(q|k_i)$  の求め方について説明する。オクルージョンが起きたときの  $\Pr(q|k_i)$  は不明であるため、その推定値  $\hat{P}(q|k_i)$  としてオクルージョンが起きていない時の  $\Pr(q|k_i)$  で代用する。そこで、ウィンドウが重なっていないときに、このヒストグラムはウィンドウが重なっていないときに、ウィンドウ中の対象領域の色領域のヒストグラムを求め、 $h$  ヒストグラムの総和が1となるように全ての値を対象領域の数で割る。この正規化したヒストグラムが、このフレームでの  $\Pr(q|k_i)$  となる。このヒストグラムを  $\hat{P}$  このヒストグラムは追跡対象の位置や姿勢の変化と共に変わってしまうため、毎フレームヒストグラムを計算し更新する。

ウィンドウが重なった時には以下の式で重み

$w_{k_0,q}$  を求める.

$$w_{k_0,q} = \frac{\hat{P}(q|k_i)}{\sum_{i=0}^p \hat{P}(q|k_i)}, \quad q = 1, 2, \dots, m \quad (8)$$

となる. ヒストグラムの色領域  $q$  の値が全てのウィンドウで 0 となる場合には

$$w_{k_0,q} = 0 \quad (9)$$

とする. この方法では, 予め追跡対象の色情報を取得しておく必要がない.

### 3.3 ウィンドウの位置の補正

ここでオクルージョン時のウィンドウの位置について考える. オクルージョンにより奥の追跡対象が手前の追跡対象にほとんど隠れてしまったとき, 奥の追跡対象のウィンドウの位置は手前の追跡対象の色に依存する. 例えば, 奥のウィンドウが重み付けを行った結果, 手前の人物の上半身や下半身に大きな重みを与えたとする. この時, 奥のウィンドウは上半身や下半身に移動し, ウィンドウの上や下の部分に対象領域が存在しない状態になる (図 1-a). この状態では奥の追跡対象が移動して画像中に現れたときに奥のウィンドウは奥の追跡対象を捉えにくくなってしまう.

そこで, 必要以上にウィンドウが対象から離れないように以下の操作によりウィンドウの位置を補正する.

1. ウィンドウ中の対象領域の重心及び分散を求める.
2. 式 (4) を用いてウィンドウの大きさを計算し, 仮のウィンドウを生成する (図 1-b).
3. 現在のウィンドウと仮のウィンドウの中心を比較し, 現在のウィンドウの中心側の辺が仮のウィンドウの辺と一致するように移動させる (図 1-c).

この操作によりウィンドウが追跡対象から大きくずれることを防ぐことが出来る (図 1-d).

### 3.4 ウィンドウの初期サイズの修正

オクルージョン中に奥の追跡対象が手前の追跡対象に完全に隠れてしまう場合が存在する. このとき奥のウィンドウが手前の追跡対象を囲っていない場

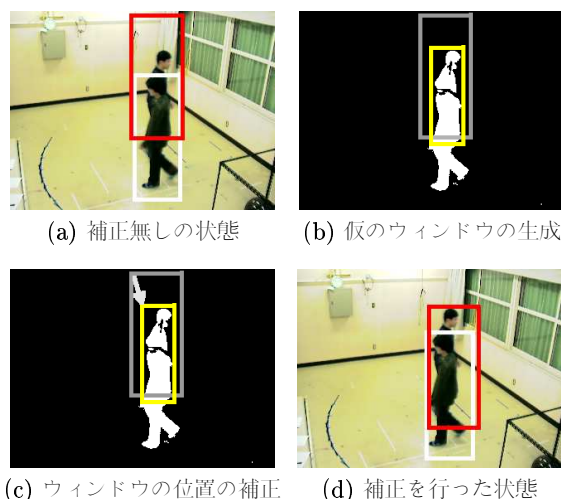


図 1: ウィンドウの位置の補正. (b),(c) では上にいる人物のウィンドウのみ表示している. 白が対象領域, 黒が背景領域である.

合には, 奥の追跡対象がウィンドウの存在しない位置から現れると追跡対象を見失う. そこで, ウィンドウの領域がある割合以上他のウィンドウと重なった場合には, 平均値シフト法を行うときのウィンドウの最初サイズを修正する. その方法としてウィンドウと重なったウィンドウの各辺を比較し, 重なったウィンドウの辺の方がウィンドウの中心から離れている場合には辺をその位置に移動させる. 平均値シフト処理の 2 回目の繰り返しからは元のウィンドウの大きさを用いる.

## 4 実験

### 4.1 実験方法

この章では, 上述した方法により実画像中の人物の追跡を行う. 本実験で用いる式のパラメータは式 (4) で  $a = 4$ ,  $b = 10(\text{pixel})$  とし, 式 (1) のパラメータを  $\alpha = 5$ ,  $\beta = 50$  とした. 平均値シフト法の繰り返しを終了させるウィンドウの移動量の閾値を  $T_h = 2$ , 最大繰り返し回数を 10 回とした. 色空間は明度  $V$  を  $0 \sim 1$  とし,  $V < 0.01$  を 1 つの色領域に  $V \geq 0.01$  を色相を 36 等分, 彩度を 5 等分し, 計 181 個の色領域を用いた. また, ウィンドウ領域の 8 割が他のウィンドウと重なった場合に初期ウィンドウサイズの修正を行った.

本実験では, SONY 製のカメラ DFW-VL500 を用いて画像を取得した. 取り込む画像の大きさを  $320 \times 240\text{pixel}$ , 画像を約 30fps で取り込んだ. 使

用したコンピュータは Pentium4-1.7GHz, OS は Linux2.4.6 である. 実験は室内(図2)で行い, 背景画像には画像 100 枚の平均を用いた. 追跡する人物の数を 3 人とし, それぞれの人物は画像の左, 下, 右から入り, 中央で交差するように歩いた. 本手法のリアルタイムでの有効性を検証するために対象人物の追跡処理にかかった時間をフレームごとに計測する.



図 2: 実験場所

#### 4.2 実験結果

追跡結果を図3に示す. また, 開始時の位置で左から順に人物 1, 2, 3 とし, 人物 1~3 の重み及び重みを計算するのに用いた色領域のヒストグラムを図4に示す. グラフの横軸の色番号  $N_c$  は色領域につけた番号であり,

$$N_c = \begin{cases} N_s \times 36 + N_h & (\text{if } V \geq 0.01) \\ 181 & (\text{if } V < 0.01) \end{cases} \quad (10)$$

としている. ここで,  $N_h$  は色相を 36 個に分割して値が小さい方から 0~35 の番号をつけたもので,  $N_s$  は彩度を 5 個に分割して値が小さい方から 0~4 の番号をつけたものである.

また, 追跡処理にかかった時間及び平均値シフト法の繰り返し回数を図5に示す. ここで, 繰り返しの回数は 3 人に対して行った回数の和である. また, 図5に表示したフレームで他のウィンドウと重なっているウィンドウの数が同じフレームでの処理時間と繰り返し回数の平均値を表1に示す.

図4の左列より, 本実験で追跡した 3 人の人物共に色番号が 156 となる色が多く存在していることが分かる. 特に, 人物 1 と人物 2 では他の色よりもその割合ははるかに大きくなっており, 2 人に存在する色が似ている. しかし, この色領域のヒストグラムから求めた重みは色番号が 156 で大きくなく, 他

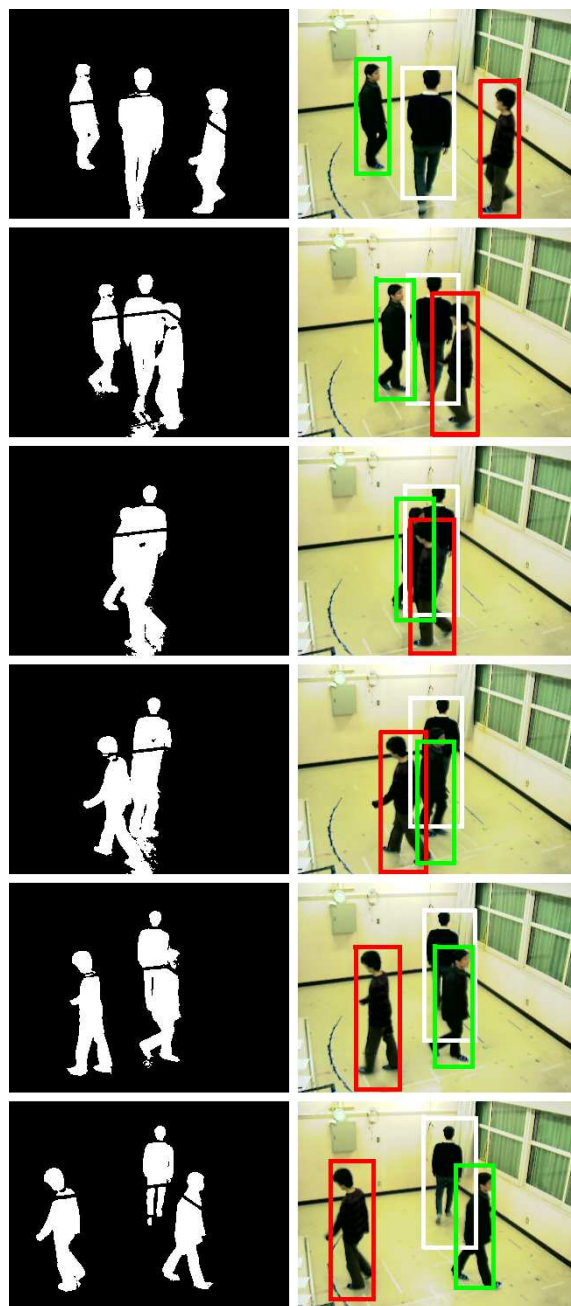


図 3: 追跡結果. 左列: 背景差分画像 (白: 対象領域, 黒: 背景領域). 右列: ウィンドウを表示した画像. 各行は画像系列を示す (上からフレーム番号 105,124,138,150,162,175).

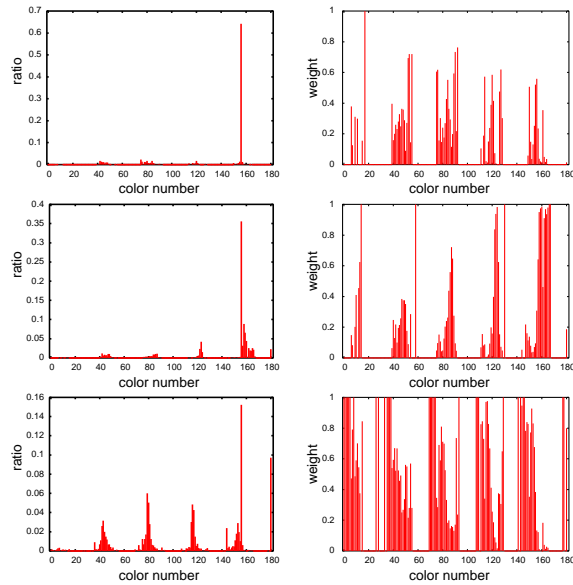


図 4: オクルージョン中の人物の色領域のヒストグラム及び重み。上から順に人物 1, 2, 3 である。

の色が大きくなるために、2 人を区別して追跡することができた。また、3 人の追跡人物が重なった場合にもそれぞれの追跡人物を区別して追跡することが確認できた。

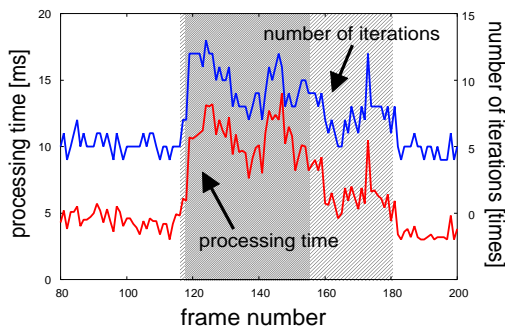


図 5: 処理時間及び繰り返し回数。背景に斜線が引かれているフレームではウィンドウ同士が重なっている。重なっているウィンドウの数は密度が小さい部分で 2 つ、密度が大きい部分で 3 つである。

図 5 よりウィンドウが重なっていないときよりも重なっているときの方が平均値シフト法の繰り返しの回数が多く、処理時間が長くなっていることが分かる。その原因として、重なっているときには他の対象領域も影響して 1 回のウィンドウの移動距離が短くなることや、ウィンドウの位置の補正を行うことによりウィンドウの移動距離の合計が大きくなる

表 1: 処理時間及び繰り返し回数の平均値

ウィンドウの数 (個)	フレーム数 (フレーム)	平均処理時間 (ms)	平均繰り返し回数 (回)
0	57	4.14	5.09
2	27	6.53	7.37
3	37	10.72	9.83

1 列目は他のウィンドウと重なったウィンドウの数である。

ことが考えられる。処理時間が長い全てのウィンドウが他のウィンドウと重なっている時でも 1 フレーム当たりの処理時間の平均値は 10.72ms(図 1 であり、平均で 1 秒間に約 93 フレーム処理できる。本実験で用いたカメラは 30fps なのでリアルタイムでの追跡が十分可能であると言える。また、処理時間は追跡人物の数や画像中の対象人物の大きさによって変化するが、本実験の結果から人数の数が倍でもリアルタイムでの追跡が可能であると考えられる。

## 5 まとめ

本論文では画像中の複数人物をリアルタイムで追跡する方法を示した。本研究では、背景差分法により抽出した対象領域に平均値シフト法を用いることにより対象人物の追跡を行った。オクルージョンが起こった場合には対象領域に重みを与えて平均値シフト法を行うことによりそれぞれの対象人物を区別して追跡した。実験により追跡人物が 3 人の場合にオクルージョンが起こっても 3 人が重なった場合にもそれぞれを区別して追跡できることを示した。また、リアルタイムでの追跡が十分可能であることも示した。

## 参考文献

- [1] Dorin Comaniciu, et al.: Real-Time Tracking of Non-Rigid Objects using Mean Shift, IEEE CVPR 2000
- [2] Gary R. Bradski: Computer Vision Face Tracking For Use in a Perceptual User Interface, Technical REport Q2, Intel Technology Journal (1998),