

# 多スタート局所探索法を用いた多数人物追跡のための 複数能動カメラの同時視線プランニング

武村 紀子\*<sup>1</sup> 三浦 純\*<sup>2</sup>

## View Planning of Multiple Active Cameras for Tracking Many Persons based on Multi-start Local Search

Noriko Takemura\*<sup>1</sup> and Jun Miura\*<sup>2</sup>

This paper describes a view planning of multiple cameras for tracking multiple persons. View planning of cameras is a very important problem in watching multiple persons in a wide area by using a few cameras. We select fixation points of cameras so that the expected number of tracked persons is maximized, based on a probabilistic model of person motion. We propose a multi-start local search-based method for tracking persons intermittently using a criterion which allows frequent shifts of fixation points. This view planning outperforms the others and is considered to be appropriate for wide area surveillance systems. We then modify the method so that the planning cost is reduced. We divide the cameras into mutually independent groups based on relations between their viewing directions and determine fixation points within each group. The performance of this modified method is comparable to the original one with a lower planning cost.

**Key Words:** View Planning, Tracking, Multi-start Local Search

### 1. はじめに

視覚監視はコンピュータビジョンにおいて活発に研究されている分野の一つである。過去のほとんどの研究は、画像処理を用いて目標物を追跡する手法の開発を目的としている [1]~[3]。本論文では、視覚監視の問題を画像処理ではなく、カメラの視線プランニングの観点から扱う。

現在、監視カメラは固定カメラを用いている場合が多く、その場合、広い空間を監視するためには、その広さに応じてカメラの設置台数を増やす必要がある。そこで、効率の良い監視システムを作るためには、視線を移動させることができるアクティブカメラの利用が有効である。その際、比較的少数のアクティブカメラを用いて広い空間を監視するためには、カメラの視線プランニングが非常に重要な問題となる。

カメラやセンサの視線プランニングについて、様々な研究がなされているが、それぞれ異なったプランニングの問題を扱っている。

青木らはネットワーク接続された計算機とアクティブカメラ

の組の論理モデルである、能動視覚エージェント群の協調動作による複数対象の実時間追跡システムを提案している [4]。この研究では、カメラ群を対象物を注視追跡するカメラ群と観測領域を撮影するカメラ群とに分けて考えている。また、Karuppiahらは対象の位置・速度などの状況に応じてカメラの選択を行う監視システムを提案している [5]。これらの研究では、比較的小さい領域内を移動する少数人物の追跡問題を扱っている。

Horlingらは分散センサネットワークを用いた協調監視システムを提案している [6]。また、Islerらは目標物の位置予測の期待誤差が最小となるように、目標物をそれぞれカメラに割り当てる手法を提案している [7]。これらの研究は、カメラの数が追跡する人数より比較的多い場合について扱っている。

Jungらは複数の目標物を追跡する複数の移動ロボットの協調監視システムを提案している [8]。この研究では、監視領域内においてどの場所が監視する必要性が高いかを計算し、それに基づいてロボットを配置させている。監視の必要性の評価は現在の目標物の分布に基づいて行われており、未来の状態については考慮されていない。

Miuraらは多数カメラによる多数人物の追跡問題をプランニングと行動の並列化という観点から扱っている [9]。ここでは、プランニングを人物のカメラへの割り当てを繰り返し改善する任意時間アルゴリズムで定式化している。また、カメラ注目点は、予測された1ステップ後の状態に基づき決定している。そのため、近視眼的な行動を選択する可能性が高いと考えられる。

原稿受付 2007年4月4日

\*<sup>1</sup>大阪大学大学院工学研究科機械工学専攻

\*<sup>2</sup>大阪大学大学院工学研究科、現：豊橋技術科学大学

\*<sup>1</sup>Department of Mechanical Engineering, Graduate School of Engineering, Osaka University

\*<sup>2</sup>Graduate School of Engineering, Osaka University, Currently with Toyohashi University of Technology

Krishna らは複数センサ監視システムにおける視線プランニング手法を提案している [10]. ここでは, 計算量の増大を避けるために, 人物の分布に基づいてセンサに優先順位をつけ, 一つずつ順に視線を決定している. その際, 個々の人物の予測は行わず, 統計的に求められた人物の分布を用いている. そのため, 個々の人物の細かい動きに対応することが困難であると考えられる.

本研究は, 不確かさを考慮した人物の位置予測に基づいた多数人物の追跡のためのカメラの視線プランニング手法の開発を目的とする. ここでは, できるだけ多くの人物を, 隣り合うカメラの視野をあまり重複させることなく効率良く捕捉するために, 全カメラの視線を一度にプランニングする. しかし, 全カメラの視線を一度にプランニングすると, 探索空間の増大に伴って計算コストが高くなるため, 多スタート局所探索法を用いて計算コストの低減を図る.

2章では, 多数カメラによる多数人物の追跡問題を定義し, 人物の動きモデルおよび人物の状態の推定方法, カメラ視線の評価方法について述べる. 3章では, 一度にカメラにとらえられる人数が多くなるようにカメラ視線をプランニングする手法について述べる. 4章では, プランニング時に捕捉されている人物よりも, 少しの間捕捉されていない人物を優先して追跡するようなプランニング手法について述べる. 5章では, 4章のプランニング手法の計算コストを下げるために, 全カメラを互いに独立ないくつかのグループに分けて, グループごとにプランニングを行う手法について述べる.

## 2. 多数カメラによる多数人物の追跡問題

### 2.1 問題設定

天井に等間隔に設置された複数のカメラを用いて, 部屋をランダムに歩きまわる複数の人物を観察するという問題を扱う. なお, ここでは, 同程度に重要な人物が多数移動している状況で, できるだけ多くの人物の動きを知ることを追跡の目的としており, 特定の人物をズームアップしてより多くの情報を得ることは考えていない.

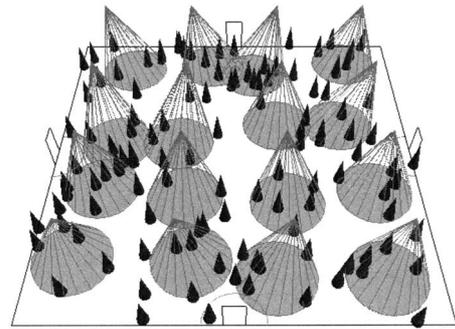
問題を簡略化するために, 以下の仮定を設ける. 障害物のない部屋高くにカメラを設置して, 人物の隠蔽は起こらないものとする. また, 人物の識別も可能であるとし, 1回の観測では, 観測時にカメラの視野内に存在する人物の位置と速度の情報が得られるとする.

カメラの視線が地面と交わる点をカメラの注目点とし, 視線プランニングは次ステップのカメラ注目点を求めることとする. また, カメラの注目点位置の候補は注目点の動ける範囲を格子状に離散化したものに限定する.

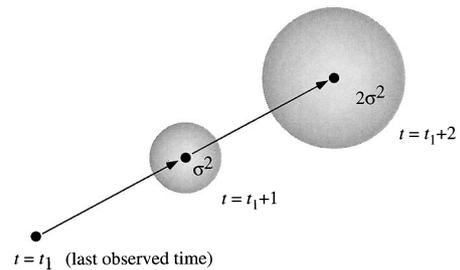
### 2.2 シミュレーション環境

**Fig. 1** に作成したシミュレータによるシミュレーションの様子を示す. ここでは, カメラの台数が4台の場合と16台の場合について考える. カメラが4台(2×2配列)の場合は, 部屋の大きさを50[m]×50[m], 部屋内の人数を30人とする. カメラが16台(4×4配列)の場合は, 部屋の大きさを100[m]×100[m], 部屋内の人数を120人とする.

カメラの視野はカメラ注目点から半径10[m]の円内, カメラ



**Fig. 1** MCMP (Multi-Camera Multi-Person) simulator. Small black cones and circles on the floor indicate persons and fields of view of cameras, respectively



**Fig. 2** Motion uncertainty model of person

の注目点の動ける範囲は初期注目点から半径10[m]の円内であるとする. また, カメラ注目点の動ける範囲を離散化する際の格子の大きさは1[m]×1[m], カメラ注目点の最大移動速度は2.5[m/s]であるとする. また, 観測周期は1[s]とする. 天井高くに設置したカメラから広範囲を観測するという設定なので, 画像内の人物の動きはそれほど速くなく, 観測周期は1[s]で十分と考えられる.

100ステップの人物の移動データをカメラ4台, 16台の場合について, それぞれ10セット, 5セット生成し, プランニング手法の評価に用いた.

### 2.3 人物の動きのモデル化と位置予測

現在捕捉されている人物は現時点での位置と速度の情報が得られるので, 信頼性のある位置予測ができる. また, 現在捕捉されていない人物で, 少し前に捕捉されていた人物についても, 予測はある程度信頼できると考えられる. そこで, ある人物が最後に捕捉されてから一定期間の間は, その人物の位置予測を行い, プランニングの際に考慮することにする.

人物の動きは等速直線運動に正規分布の誤差が加わるものとし, 観測時に得られる人物の位置と速度の情報をもとに人物の位置予測を行う. ある人物の, 最後に捕捉された時刻から $t$ ステップ後の予測位置は, その時刻での位置と速度で等速直線運動によって $t$ ステップ後に到達する位置を中心に分散 $\sigma^2 t$ をもつ等方正規分布に従うと考える (**Fig. 2** 参照). 人物はいわゆる $3\sigma$ の領域内に存在するとする. ある一定ステップ前までに捕捉された人物それぞれについて位置予測を行うと, 予測位置の確率分布の集合が得られ, それを基にプランニングを行う.

### 2.4 人物の動きの予測に基づくカメラ視線の評価

プランニングを行う際は, 現時点から一定ステップ後までの

人物の動きを予測し、一定ステップ後までの総捕捉人数の期待値が最大となるようにカメラ視線の動きを考えた上で、1ステップ後のカメラ注目点位置を決定する。捕捉人数の期待値は、カメラの視野と人物の確率分布の円が重なる部分の確率を足し合わせることによって計算する。総捕捉人数の期待値が最大となるカメラの注目点位置が複数存在する場合は、以下の(1)の評価基準を用いて評価し、それでも評価が等しいものが複数存在する場合は、(2)の評価基準を用いて評価する。

- (1) カメラ注目点の移動距離が最小：カメラの注目点の動きが少なくなり、滑らかに動くようになる。
- (2) カメラの初期注目点からの距離が最小：カメラの注目点が初期位置から大きくずれるのを防ぐことができる。カメラ注目点が初期位置付近にある場合、あらゆる方向に注目点を動かすことができるので、様々な状況に対応することができる。また、初期注目点はカメラ同士の視野が重ならないように等間隔に配置されているので、カメラの視線をなるべく均等に分散することができる。

### 3. 連続的な追跡のためのプランニング

#### 3.1 解の探索

全カメラの注目点の組み合わせを解として、解の探索を行う。全ステップを通して、カメラに捕捉される平均人数が最大となるようなプランニングを行うのが理想であるが、何ステップも先の人物の位置を予測するのは難しい上に、計算時間が膨大になるので、そのようなプランニングは非常に困難である。そこで、現在から $T$ ステップ後までの総捕捉人数の期待値が最大となるように解の探索を行う。

捕捉されていない期間が $T$ ステップ未満の人物について位置予測を行い、2.4節の評価基準に基づいて、現在から $T$ ステップ後までのカメラ注目点を順に決定していく。各ステップにおける評価値を足し合わせて、 $T$ ステップ分のカメラの注目点候補の組み合わせのうち、最も評価が良くなるものを選択し、1ステップ後におけるカメラの注目点を決定する。

解の探索に全探索法を用れば、すべての解について評価を行うので良い解が得られる。しかし、探索空間は全カメラのとり得るすべての注目点の組み合わせとなり、探索時間が非常に長くなる。そこで、短い時間で解の探索を行うために局所探索法を用いる。ここでは、局所探索法の中でも比較的アルゴリズムが単純で探索時間が短いと考えられる多スタート局所探索法 (MLS法) [11] を用いる。

#### 3.2 多スタート局所探索法 (multi-start local search, MLS法)

多スタート局所探索法 (MLS法) は、探索におけるメタ戦略のうち、初期解生成に多スタート法を用い、局所探索には単純探索法を用いる方法の総称で、メタ戦略の中では古くから広く用いられてきた [11]。MLS法は、

- (1) ランダムに、または欲張り法を用いて初期解  $x$  を生成し、
  - (2) 解  $x$  を単純局所探索法により改善する、
- という二つのステップを反復し、この過程で得られた最良の実行可能解を出力するという方法である。

ここでは、解の近傍として、ある一つのカメラの注目点をそ

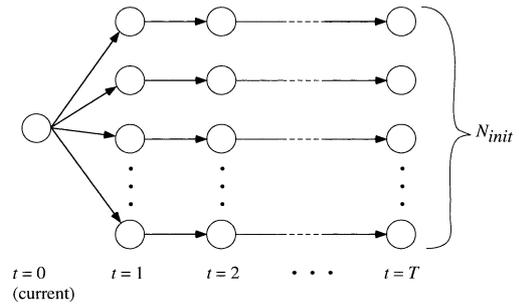


Fig. 3 Search tree

の8近傍のいずれかに移動させたものを用いる。全カメラの注目点の組み合わせを解としているため、カメラが $N$ 個ある場合は、近傍は $8N$ 個存在する。

#### 3.2.1 探索方法

1ステップ後に取り得るカメラの注目点の組み合わせから、初期解をある一定の数だけランダムにとる。それぞれの初期解について、2.3節の評価基準に基づき局所探索を行って局所最適解を求め、1ステップ後の解集合とする。次に各解についてさらに1ステップ後に取り得る注目点の範囲を求め、その範囲内でMLS法により局所最適解を求め、2ステップ後の解集合とする。以上を $T$ ステップ後まで繰り返す (Fig. 3参照)。以上より求めた $T$ ステップ分の解の組み合わせのうち、 $T$ ステップ後までの解の評価値の和が最も良いものを選択し、1ステップ後のカメラの注目点を定める。

#### 3.3 シミュレーション実験

Xeon 3GHzのPCを用い、カメラ4台の場合についてシミュレーション実験を行った。

##### 3.3.1 評価法

プランニング手法の評価は平均捕捉率と平均実行時間を用いて行う。平均捕捉率とは、プランニングを行うことにより求めたカメラの注目点に対して、実際に何%の人物が捕捉されたのかを各ステップごとに計算し、全ステップ分の平均をとったものである。

##### 3.3.2 MLS法における初期解の数の決定

MLS法を用いた場合、平均捕捉率は初期解の数に依存する。何ステップ先まで人物の位置予測を行うかによって、適切な初期解の数は変化する。よって、 $T$ の値ごとに初期解の数を実験的に決定した。 $T=2$ の場合の初期解の数と平均捕捉率との関係はFig. 4のようになった。よって、 $T=2$ の場合の初期解の数を15とする。 $T=1$ 、 $T=5$ の場合も同様に、初期解の数と捕捉率との関係を求め、初期解の数をそれぞれ5、15とした。

##### 3.3.3 MLS法と全探索法との比較

$T=1, 2, 5$ について、MLS法と全探索法の比較を行った。ただし、全探索では $T > 2$ の場合、探索空間が非常に大きくなり、解が求められなかった。全データの平均捕捉率の平均値をTable 1に、1ステップ当たりの平均実行時間をTable 2に示す。

全探索法とMLS法について、最も結果が良かったもの同士を比較すると、(MLS: $T=5$ )/(全探索: $T=2$ )=0.997となった。また、全探索法の場合に対してMLS法がはるかに実行時間が

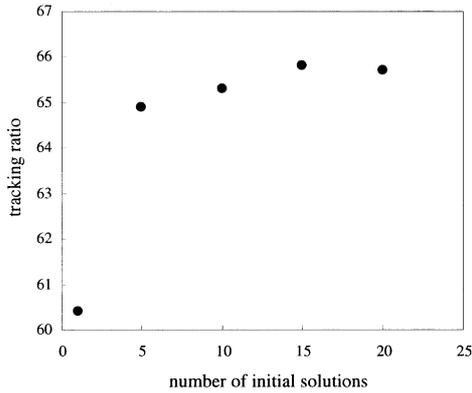


Fig. 4 Number of initial solutions vs. tracking ratio

Table 1 Tracking ratio

	$T = 1$	$T = 2$	$T = 5$
<i>exhaustive</i>	66.6%	67.5%	-
<i>MLS</i>	65.3%	65.8%	67.3%

Table 2 Calculation time per step

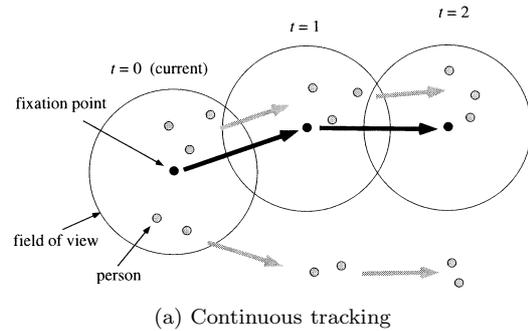
	$T = 1$	$T = 2$	$T = 5$
<i>exhaustive</i>	0.23 s	143.62 s	-
<i>MLS</i>	0.02 s	0.03 s	0.07 s

短くなった。以上より、MLS法を用いた場合は、全探索法に近い結果が得られると同時に大幅に時間が短縮できている。よって、今回の問題における解の探索法として、提案手法は適切な方法である。

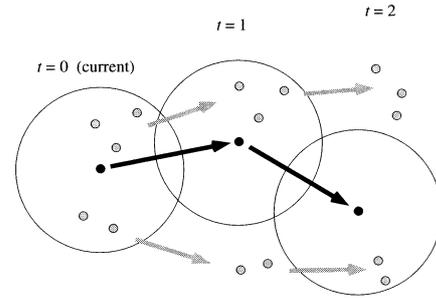
#### 4. 断続的な追跡のためのプランニング

カルマンフィルタやパーティクルフィルタなどの統計的データの統合に基づく物体追跡では（文献[12][13]など）、確率的な状態遷移モデルを用いているため、観測が得られない場合でもある程度物体位置を予測できる。このような物体追跡手法の存在を仮定すれば、ある人物が捕捉されていないときでも、捕捉されていない期間が十分短かければ、その人物の現在の位置を容易に予測できるので、その期間も捕捉されていたと見なしてよい。これにより人物を見失うことなく視線を様々な方向へ動かすことが可能になり、結果として捕捉率が向上する。

現在は捕捉されていない期間が2ステップ以下の場合、視野に入っていない期間についても捕捉されていたものと見なす。評価方法を変更する前後のカメラ視線の動きの例をFig. 5に示す。3章で述べたようにカメラの視野内に存在する人物のみを捕捉人数として数える場合、Fig. 5(a)のように、同一人物を追跡し続け、ほとんど追跡されない人物が存在する可能性がある。一方、上述のように評価方法を変更した場合には、カメラの視線はFig. 5(b)のように少しの間捕捉されていない人物を優先的にとらえるような動きをすることが期待される。この場合の3ステップ間の総捕捉人数は、1ステップ間捕捉されていない2人についても捕捉されていたと見なすため、12人となり、Fig. 5(a)の11人より増加する。以上のような評価方法を用いたプランニング手法を断続的な追跡のためのプランニング手法と呼ぶ。



(a) Continuous tracking



(b) Intermittent tracking

Fig. 5 Different behaviors for different evaluation criteria

#### 4.1 探索方法

断続的な追跡のための評価方法では、ある時刻での捕捉人数の値は、人物が以前に捕捉されていたかどうかによって変化するので、前後のステップにおけるカメラの注目点位置に依存する。よって、ステップごとに解の探索を行うのではなく、一定ステップ分（現在は5ステップ分）の注目点の組み合わせを解として探索を行う。このとき、解の探索空間は、全カメラが5ステップの間にとり得るすべての注目点の組み合わせの集合となる。

解の近傍としては、ある一つのカメラのあるステップにおける注目点をその8近傍のいずれかに移動させたものを用いる。カメラが $N$ 個ある場合は、近傍は $8 \times 5 \times N$ 個存在する。

また、MLS法の初期解の数と捕捉率との関係を3章と同様に実験的に求め、初期解の数を15とした。

#### 4.2 初期解の求め方

連続的な追跡のためのプランニング（3章参照）においては、1ステップ分ずつ解の探索を行うので、局所最適解の数が比較的少なく、初期解をランダムに選んでも、ある程度評価の良い局所最適解に収束する。しかし、断続的な追跡のためのプランニングにおいては、全カメラの5ステップ分の注目点の組み合わせを解としているので、解の探索範囲が広く、連続的な追跡のためのプランニングと比較すると局所最適解の数がかなり多くなるため、ランダムに初期解を選ぶと評価の良い局所最適解に到達することが困難になる。連続的な追跡のためのプランニング結果を初期解として利用することも考えられるが、このときカメラは同一人物を追跡して、カメラ視線は滑らかに動こうとするため、断続的な追跡のためのプランニングの初期解としては適していない。そこで、初期解を評価が良いと思われる局所最適解のある程度近くに生成するために、以下の方法を提案

する。

- (1) マップの作成 (Fig. 6 (a) 参照) : それぞれのステップについて, カメラの注目点における捕捉人数を記述したマップを作成する. ただし, マップ作成時はカメラの重なりを考慮しない. ここで, 図中の正方形は人物を表し, 濃淡はカメラ注目点がある位置にあるときの捕捉人数を示している.
- (2) 代表点の決定 : それぞれのマップにおいて, そのステップまでにカメラの注目点が到達できる範囲を求め, その範囲内にある一定の大きさの領域 (現在は  $5[m] \times 5[m]$ ) に分割し, それぞれの領域内で, 捕捉人数の期待値が最も高い点をその領域の代表点とする (Fig. 6 (b) 参照). ここで, 図中の円はカメラの注目点が到達できる範囲を示し, 黒い+印は代表点を示している.
- (3) 候補点の選択 (Fig. 6 (b) 参照) : 代表点における捕捉人数の期待値があるしきい値 (現在は最大値の 90%) 以上となる代表点を候補点とする. ここで, 図中の白い+印は候補点を示している.
- (4) 初期解の決定 :  $T$  ステップ分 (現在は  $T = 5$ ) の候補点の中からランダムに一つ選ぶ. ここで, 選ばれた候補点の時間ステップを  $t_s$  とすると,  $t_s = T$  の場合,  $t = 1 \sim T - 1$  でのカメラ視線は,  $t = T$  の点まで等速直線運動をすると考えたときに通る点を定める (Fig. 7 (a) 参照). 図中の平行線は二次元のマップを横から見たものである. また,  $t_s < T$  の場合,  $t = 1 \sim t_s - 1$  でのカメラ視線は同様に補間して求める. 次に,  $t > t_s$  の候補点について,  $t_s = T$  となるまで同様の操作を繰り返す (Fig. 7 (b) 参照).

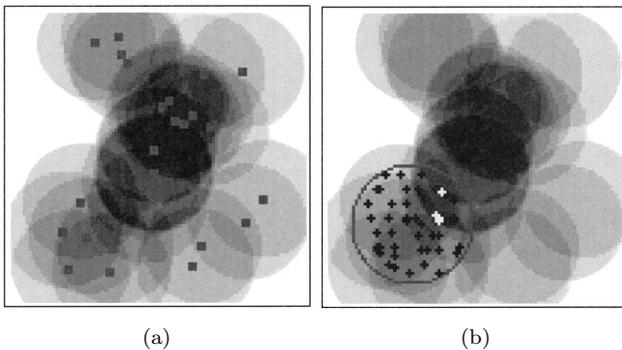


Fig. 6 Generating a map of promising fixation points at a time step

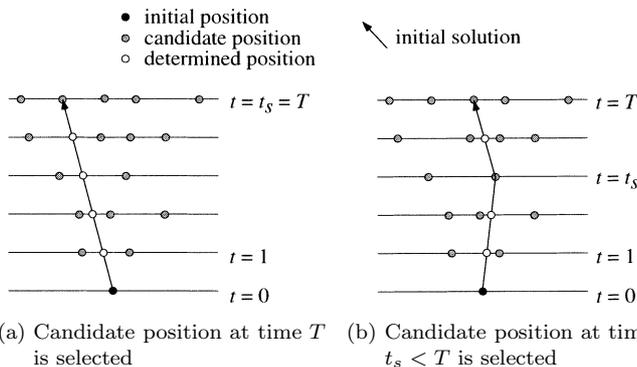


Fig. 7 Generate initial solutions

### 4.3 シミュレーション実験

Xeon 3GHz の PC15 台からなる PC クラスタを使用し, カメラ 4 台の場合について, 以下のシミュレーションを行った. 15 個の初期解それぞれに対し, 各 PC で並列に探索を行った.

#### 4.3.1 初期解生成方法の比較

以下の三つの初期解生成方法を用いた場合についてそれぞれシミュレーションを行った.

- (1) 提案手法
- (2) 連続的な追跡のためのプランニング結果を初期解とする手法
- (3) ランダムに初期解を生成する手法

Table 3 にそれぞれの初期解生成方法を用いた場合の結果を示す. Table 3 より, 提案した初期解生成方法を用いた場合の平均捕捉率が最も高くなり, 提案した方法の有用性が示された.

#### 4.3.2 他のプランニング手法との比較

以下の五つの手法についてシミュレーションを行った.

- (1) 断続的な追跡のためのプランニング手法
- (2) 連続的な追跡のためのプランニング手法 (3 章参照)
- (3) カメラ同士の視野の重なりを考慮しないで, 1 台ずつ独立にプランニングを行う手法: この場合は, 解があるカメラの 5 ステップ分のカメラ注目点の組み合わせとし, (1) と同様の方法を用いて, 各カメラについて 1 台ずつ解の探索を行う
- (4) カメラ注目点を完全にランダムに決める手法
- (5) カメラの注目点を初期位置のまま固定させる手法

プランニング手法の評価を行う際は, すべて断続的な追跡の場合と同様の評価を行った.

五つの手法の平均捕捉率を Table 4 に示す. (4), (5) の手法の結果は極端に悪いが, (1)~(3) の手法については, ある程度良い値が得られた. また, (1)~(3) の場合を比較すると, (1) の場合に最も良い値が得られた. すべてのカメラで一度に観察できる領域の大きさは, 最大でも部屋の大きさの 50% 程度であり, カメラを固定した場合は平均捕捉率が 50% に満たないという結果になった. (1)~(3) の 1 ステップ当たりの平均実行時間は Table 5 のようになり, (1) の手法の実行時間は他のものに比べて多少長くなっているが, 本手法を適用することにより約 75% まで平均捕捉率を向上させることができた.

#### 4.3.3 様々な問題設定による比較

簡単な問題設定の場合, どの手法を用いてもある程度良い結

Table 3 Comparison of methods of generating initial solutions in terms of tracking ratio

Method	(1) Proposed	(2) MLS solution	(3) Random
Tracking ratio	74.2%	72.2%	73.4%

Table 4 Comparison of five methods

(1)intermittent	(2)continuous	(3)independent	(4)random	(5)fixed
74.2%	70.8%	69.6%	48.8%	46.9%

Table 5 Computation time

(1)intermittent	(2)continuous	(3)independent
0.33 s	0.09 s	0.09 s

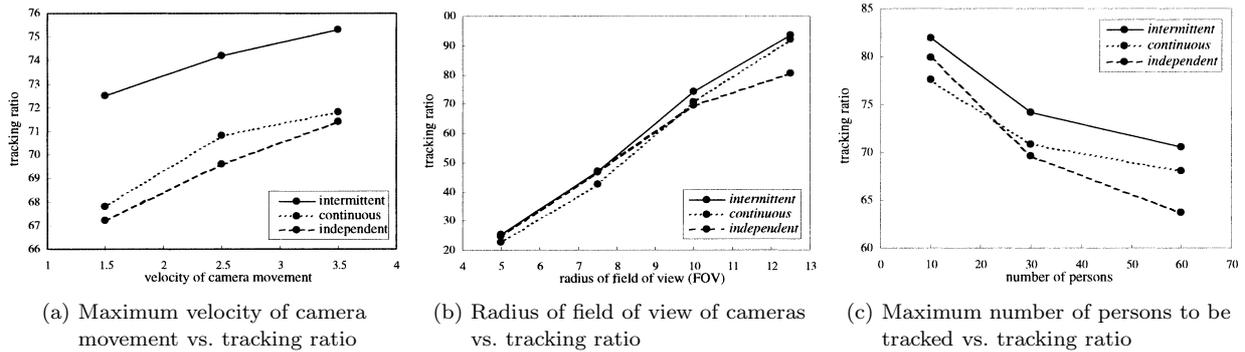
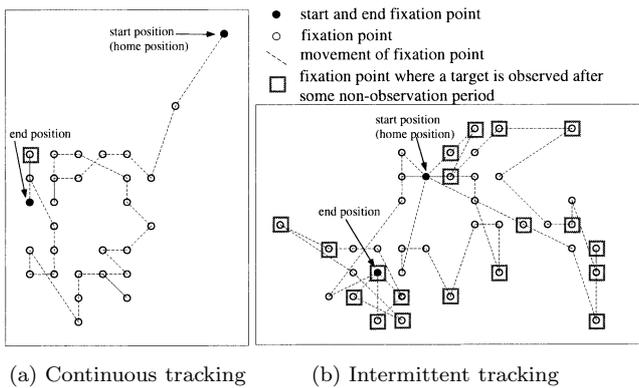


Fig. 8 Comparison in various problem settings

Fig. 9 Camera movements for *continuous* and *intermittent*. The start positions are the same actually

果が得られるが、難しい問題設定の場合、良い手法とそうでない手法との差がより明確になると思われる。そこで、(1)~(3)の手法について、カメラの最高速度、カメラの視野の大きさ、部屋内の人数をそれぞれを変化させた場合の平均捕捉率を Fig. 8 に示す。どのように環境を変化させた場合についても、(1)の場合が最も良い結果を得られた。また、問題を難しくしていったときの平均捕捉率の低下率も(1)の場合が最も小さくなった。これにより、提案した断続的な追跡のためのプランニング手法の有用性が示された。

#### 4.3.4 カメラ視線の動きの比較

ある問題において、連続的に追跡を行った場合と断続的に追跡を行った場合の100ステップ分のあるカメラの視線の動きを Fig. 9 に示す。それぞれの場合のカメラ視線の動きを比較すると、Fig. 9(a)より、連続的な場合は正方形(短期間捕捉されていない人物を捕捉した位置)が1個しかなく、同一人物を連続的に追跡していることが分かる。また、断続的な場合は、カメラ視線を様々な方向に大きく動かしているため、様々な人物を捕捉するのに適している。実際に Fig. 9 における100ステップ間の総捕捉人数を比較すると、連続的な場合は432人、断続的な場合は506人となり、断続的な場合の方がかなり良い結果が得られる。このことから提案手法の有用性が分かる。

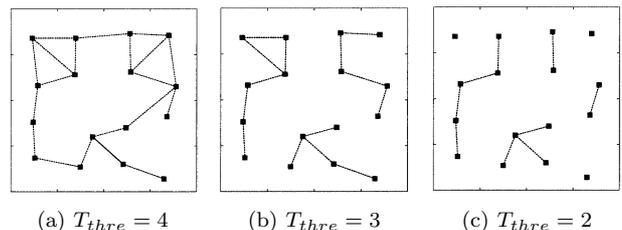


Fig. 10 Groupings for several thresholds

## 5. グループごとの解の探索

全カメラについて一度に解の探索を行うと、1台ずつ解の探索を行う場合に比べ、探索空間が広くプランニング時間が長くなる。そこで、プランニング時間を短縮するために、全カメラをいくつかのグループに分割し、グループごとに順に解の探索を行う。

### 5.1 独立グループの生成

カメラ同士的位置関係や人物の予測位置に基づき、全カメラを互いに独立なグループに分割する。考慮している期間内で、同一人物を追跡する可能性がないカメラ同士は、異なるグループに属する。

ある人物が捕捉時から  $T_{thre} (\leq T)$  ステップの間に2台のカメラによってとらえられる場合、すなわち、人物が捕捉時から  $T_{thre}$  ステップの間に2台のカメラの視野の重複部分に存在する可能性がある場合、これらの2台のカメラは互いに依存関係にあるという。すべてのカメラ同士の関係を調べ、互いに依存関係にあるカメラを同一グループに所属させる。

ある時刻において、 $T_{thre}$  を変化させたときのカメラ同士の関係の典型例を Fig. 10 に示す。図中の点と点線はそれぞれカメラの注目点位置と依存関係を示している。 $T_{thre}$  が大きいほど、グループの数は少なくなる。Fig. 11 に  $T_{thre}$  を変化させた場合の平均捕捉率と計算速度(1秒間にプランニングできるステップ数)の変化を示す。Fig. 11 より、 $T_{thre} = 2$  あるいは、 $T_{thre} = 3$  の場合、比較的短時間で、ある程度良い結果が得られる。よって、ここでは、 $T_{thre} = 3$  として、以下の実験を行う。

### 5.2 探索方法

5.1 節で述べた方法でカメラを互いに独立なグループに分割し、それぞれのグループを所属するカメラの数が多い順に優先順位をつける。優先順位の高いグループから順に、4章と同様

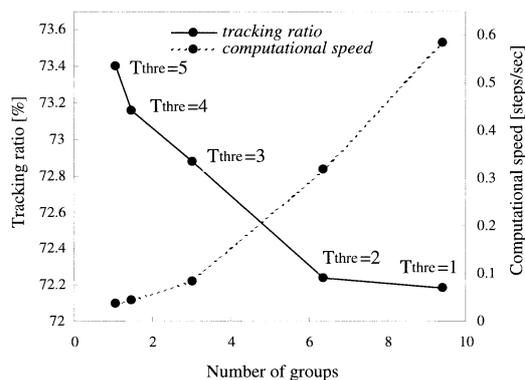


Fig. 11 Tracking ratio, computation speed and the number of groups for changing  $T_{thre}$

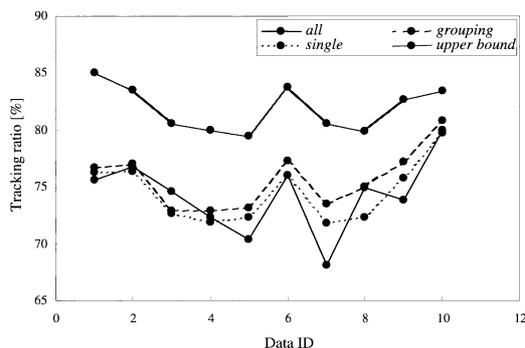


Fig. 12 Tracking ratio (Four-camera case)

の方法を用いて一度にプランニングを行う。ただし、優先順位が等しい場合はランダムに順番を決める。また、すでに解を求めたグループに属するカメラについては、求めた位置にカメラ視線を固定して、それらのカメラとの視野の重なりを考慮して、他のグループについてプランニングを行う。また、各グループに対する解は、グループ内の全カメラの5ステップ分のカメラ注目点の組み合わせとする。

また、初期解の数を4章と同様に求め、15とした。

### 5.3 シミュレーション実験

Xeon 3GHzのPC15台からなるPCクラスタを使用し、カメラ4台、16台のそれぞれの場合について、以下のシミュレーションを行った。15個の初期解それぞれに対し、各PCで並列に探索を行った。

- (a) 一度に解の探索を行う手法 (4章参照)
- (b) グループごとに解の探索を行う手法
- (c) (b)において、1台のカメラが一つのグループを生成する手法

また、(a)~(c)の均捕捉率の上限值を得るために以下の場合についてもシミュレーションを行った。

- (d) 全人物の動きが既知で、一度に解の探索を行う手法

以上の四つの場合の平均捕捉率を Fig. 12 (カメラ4台)、Fig. 13 (カメラ16台)に示す。また、(d)の平均捕捉率を1としたときの(a)~(c)の割合を Table 6に示す。また、(a)~(c)の場合における1ステップ当たりの平均実行時間を Table 7に示す。

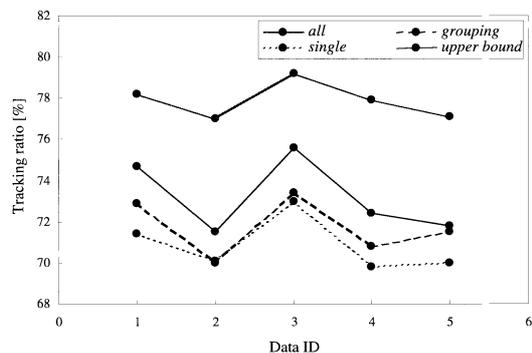


Fig. 13 Tracking ratio (Sixteen-camera case)

Table 6 Achievement ratio

	(a)all (d)upper bound	(b)grouping (d)upper bound	(c)single (d)upper bound
4-camera case	0.906	0.924	0.910
16-camera case	0.940	0.920	0.910

Table 7 Computation time

	(a)all	(b)grouping	(c)single
4-camera case	0.3s	0.2s	0.1s
16-camera case	48.0s	10.6s	1.1s

平均捕捉率は、カメラ同士の干渉と解の探索空間の大きさによって変動すると考えられる。カメラ4台の場合に対し、カメラ16台の場合は干渉し合うカメラの台数が多いので、カメラ同士の干渉が大きい。また、各手法におけるカメラ同士の干渉の考慮の度合いは、(a) > (b) > (c)となる。一方、解の探索空間が小さいほど、より良い解が見つかる可能性が高いと考えられる。したがって、カメラ4台の場合に対し、カメラ16台の場合には、解の探索空間が広がる。また、各手法における探索空間の大きさは、(a) > (b) > (c)となる。

カメラ4台の場合について、(a)~(c)の平均捕捉率を比較すると、(b)の場合が最も良く、(a)と(c)は(b)よりも少し悪い値が得られた。(a)と(c)の結果が悪くなったのは、(a)では解の探索空間の大きさによる影響が、(c)ではカメラ同士の干渉による影響が大きかったためであると考えられる。

また、カメラ16台の場合について、(a)~(c)の平均捕捉率を比較すると、(a)、(b)、(c)の順に良い結果が得られた。これは、本手法では、MLS法を用いて解の探索を行う際に、初期解の生成方法を工夫して探索空間内の有用な解のありそうな部分だけ探索しているため、カメラの台数の増加による探索空間の大きさの影響の増加の差はある程度抑えられ、その結果、カメラ同士の干渉による影響のほうが強くなったと考えられる。

また、どの手法を用いた場合も、平均捕捉率が理想的な場合の90%を超えている。平均実行時間については、(c)の場合が最も短くなった。また、(a)に対して、(b)はかなり時間が短縮されている。

(b)の結果はグループ化のためのしきい値  $T_{thre}$  を大きくすれば(a)の結果に、小さくすれば(c)の結果に近づくと考えられる。グループ化のためのしきい値として実際にどの値を用いればよいかは、対象の速度や計算機性能など様々な条件に依存

するが、以上の実験結果から提案するプランニング手法によって、追跡の精度とプランニング時間を制御することができることが示された。

## 6. おわりに

本稿では、多数人物追跡のための複数能動カメラの視線プランニング手法として、多スタート局所探索法を用いて断続的に人物の追跡を行うための視線を効率的に決定する手法を提案した。さらに、プランニング時間を短縮するために、カメラ群を互いに影響の小さいグループに分割し、グループごとに解の探索を行う方法を提案した。また、これらの手法の有用性を実験的に確かめた。

現在は、人物同士の隠蔽は生じず視野内の人物は確実に観測できる、一度の観測で人物の速度・位置の情報が得られる、人物の認識が可能であるといった、様々な仮定の下でプランニングを行っている。より現実的な仮定の下で有効なプランニング手法の開発が今後の課題である。例えば、隠蔽が生じる場合には、人物の位置だけでなく隠蔽可能性も含めた予測や、部分的な隠蔽状態における捕捉の評価が必要になると考えられる。また、上記の仮定を満たすような環境で、実際に取得した人物の動きデータに対して、本手法を適用させることも今後の課題である。

## 参考文献

- [1] H. Buxton: "Learning and understanding dynamic scene activity: a review," *Image Vision Comput.*, vol.21, no.1, pp.125-136, 2003.
- [2] L. Lee, R. Romano and G. Stein: "Monitoring activities from multiple video streams: Establishing a common coordinate frame," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.22, no.8, pp.758-767, 2000.
- [3] C. Stauffer, W. Eric and L. Grimson: "Learning patterns of activity using real time tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol.22, no.8, pp.747-757, 2000.
- [4] 青木, 吉田, 荒木, 浮田, 木戸出: "アクティブカメラ群による全観測領域の協調監視の評価", *情報処理学会論文誌, コンピュータビジョンとイメージメディア (CVIM)*, vol.48, no.SIG17, pp.65-77, 2007.
- [5] D. Karupiah, R. Grupen, A. Hanson and E. Riseman: "Smart resource reconfiguration by exploiting dynamics in perceptual tasks," *Proceedings of the 2005 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp.1854-1860, 2005.
- [6] B. Horling, R. Vincent, R. Mailler, J. Shen, R. Becker, K. Rawlins and V. Lesser: "Distributed sensor network for real time tracking," *Proceedings of the 5th Int. Conf. on Autonomous Agents*, pp.417-424, 2001.
- [7] V. Isler, S. Khanna, J. Spletzer and C. Taylor: "Target tracking with distributed sensors: The focus of attention problem," *Computer Vision and Image Understanding*, vol.100, no.1-2, pp.225-247, 2005.
- [8] B. Jung and G. Sukhatme: "A generalized region-based approach for multi-target tracking in outdoor environments," *Proceedings of the 2004 IEEE Int. Conf. on Robotics and Automation*, pp.2189-2195, 2004.
- [9] J. Miura and Y. Shirai: "Parallel scheduling of planning and action for realizing an efficient and reactive robotic system," *Proceedings of the 7th Int. Conf. on Control, Automation, Robotics and Vision*, pp.246-251, 2002.
- [10] K. Krishna, H. Hexmoor and S. Sogani: "A t-step ahead constrained optimal target direction algorithm for a multi sensor surveillance system," *Proceedings of the 2005 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp.1840-1845, 2005.
- [11] 柳浦, 茨木: *組み合わせ最適化—メタ戦略を中心として—*. 朝倉書店, 2004.
- [12] S. Maskell, M. Rollason, N. Gordon and D. Salmond: "Efficient particle filtering for multiple target tracking with application to tracking in structured images," *Image Vision Comput.*, vol.21, no.10, pp.931-939, 2003.
- [13] H. Koyasu, J. Miura and Y. Shirai: "Realtime omnidirectional stereo for obstacle detection and tracking in dynamic environments," *Proceedings of the 2001 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp.31-36, 2001.



武村紀子 (Noriko Takemura)

1982年9月16日生。2006年大阪大学工学部応用理工学科卒業。同年同大学大学院工学研究科機械工学専攻博士前期課程入学。現在に至る。IEEEの学生会員。(日本ロボット学会学生会員)



三浦 純 (Jun Miura)

1984年東京大学工学部機械工学科卒業。1989年同大学院工学系研究科情報工学専攻博士課程修了、工学博士。同年大阪大学助手。同大学大学院工学研究科機械工学専攻助教授を経て、2007年豊橋技術科学大学情報工学系教授。知能ロボット、人工知能、コンピュータビジョンの研究に従事。1994~1995年CMU客員研究員。1997年ロボット学会論文賞受賞。人工知能学会、電子情報通信学会、情報処理学会、システム制御情報学会、日本機械学会、IEEE、AAAI各会員。(日本ロボット学会正会員)