

K-Shortest Paths を用いた多人数追跡におけるデータ削減による高速化

Performance enhancement of multi-human tracking based on K-Shortest Paths by data reduction

秦 希望^{*1}
Nozomi Hata

西川 由理^{*3*4}
Yuri Nishikawa

中山 俊^{*1}
Takashi Nakayama

小澤 順^{*3}
Jun Ozawa

藤澤 克樹^{*2*3}
Katsuki Fujisawa

^{*1}九州大学大学院 数理学府

Graduate School of Mathematics, Kyushu University, Japan

^{*2}九州大学 マス・フォア・インダストリ研究所

Institute of Mathematics for Industry, Kyushu University, Japan

^{*3}産業技術総合研究所

National Institute of Advanced Industrial Science and Technology, Japan

^{*4}パナソニック株式会社

Panasonic Corporation, Japan

Object tracking is a challenging problem and it has been improving dramatically in recent years. In this paper, we perform parallelized multi-object tracking system. Object tracking problem has 2 difficulties; one is to detect objects collect, and the other is to track collect using the collect object detection. Jerome et al. performed a multi-object tracking system using K-Shortest Paths to avoid these problems efficiently. However, it is difficult to calculate in parallel because of the iterations calculation of shortest paths on the graph while changing the weight of graph. In our method, we divided time intervals to apply KSP method from Probability Occupancy Map(POM), which is also obtained via using KSP method. Performance evaluation shows our algorithm is 5.4 times faster than the original KSP with 87% accuracy.

1. はじめに

カメラやビーコンなどのデータを用いた人物の検出・追跡に関する研究は現在も盛んである。人物応用として、監視カメラの映像内での人物検出による防犯への活用や、また商業施設内における売上と客の分布の関係を見ることによってマーケティングに役立てることができるなど、その活用方法は多岐に渡っている。

この中で動体の検出については、機械学習を用いることで人や車など検出対象の画像を学習した上で検出する方法が代表的である。動画の画像情報のみから追跡を行うことは難しく、特に多数の人物が入り混じっているような状況下での正確な検出は非常に困難であり、ID の入れ替わりや消失が起こることが多い。

上記を解決するための先行研究として、機械学習で出力された検出結果を入力とする K-Shortest Paths(KSP) を用いたものが提案されている [1]。KSP を用いた方法では図 1 のような手順を取る。対象領域上に点を配置したグラフを作成した上で、まず機械学習を用いた人物検出の結果から各地点ごとの人物存在確率 (Probability Occupancy Map: POM) [2] を求め、POM を基にした枝長を持つ時間拡大したネットワーク上で人数分の最短路を解くことで人物追跡を行っている。このことによって一時的な多重検知・未検知等の誤差を修正しながら人物の追跡を行うことができる。

本研究は、この手法を適用すべき場面を POM から選択することで KSP の適用範囲を局所的なものとし、計算の高速化・並列化を目的としたものである。KSP 内では追跡対象とする動画の撮影範囲・時間幅に応じた大きさのグラフ上で繰り返し最短路を解いており、KSP そのものを並列化することは難しい。そこで本研究では、混雑が起きているかどうかを POM から判断し、その結果を KSP の適用対象かどうか用いること

で人物追跡の並列化を行い、計算時間を短縮した。人物検出の精度には様々な要因が関連して来るが、画像内の人々の混雑度合いはその要因の一つと考えられる。一般に、混雑が発生している場合はそうでない場合と比較して検出の精度が落ちる。検出の精度が高い部分で KSP を分割することで、分割した際の整合性を保ちつつ計算の分割を行うことが可能となる。また分割方法は POM にのみ依存しているため、本追跡・分割手法はカメラやビーコンなどの位置取得の方法によらず、汎用的な適用が可能である。

2. KSP による動線追跡方法

動線追跡方法として用いた KSP による手法 [1] では、人物検出を行った後、整合性が取れる様に検出結果を繋ぐことによって動線追跡を実現している。人物検出に際しては対象領域を撮影した複数の映像を入力として用いており、各映像内の各時刻における検出結果を地点ごとの存在確率として出力しておく。出力の際には、図 2 のようにあらかじめ検出領域を矩形領域に各矩形内の人数が高々1人となるよう十分小さく分割した上で、各矩形領域内の存在確率を出力する。

2.1 問題設定

分割された矩形領域の個数を K 、解析対象とする映像のフレーム数を T とする。矩形領域の大きさ及び 1 秒あたりのフレーム数を考慮し、時刻 t において領域 k に存在した人物は次の時刻 $t+1$ で k に隣り合う領域 $\mathcal{N}(k) \subset \{1, \dots, K\}$ に移動できるものとする。このとき対象領域のグラフ $G = (V, A)$ が

$$V := \{1, \dots, K\}$$

$$A := \{(i, j) \in V^2 \mid i = 1, \dots, K, j \in \mathcal{N}(i)\}$$

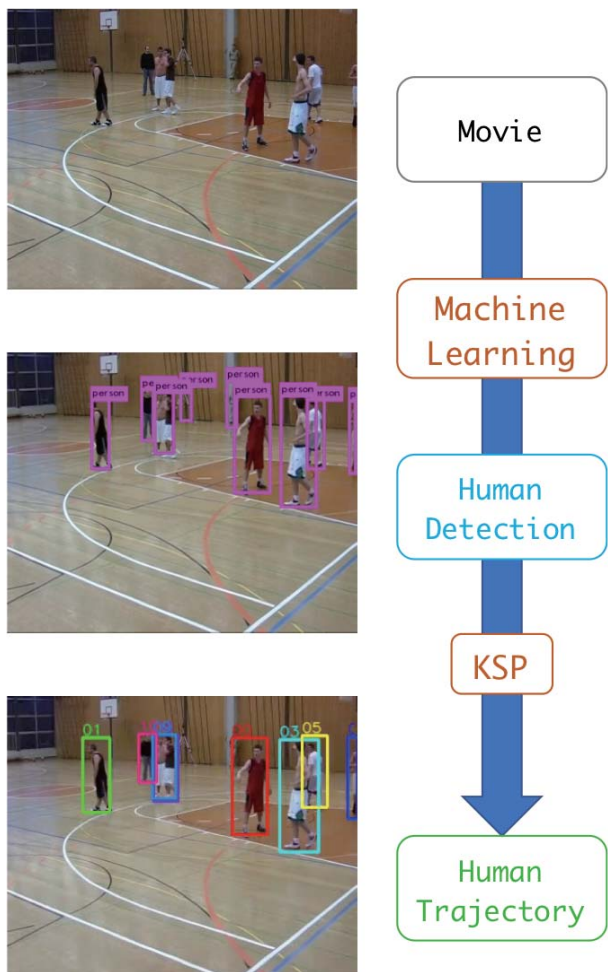


図 1: KSP による動線追跡

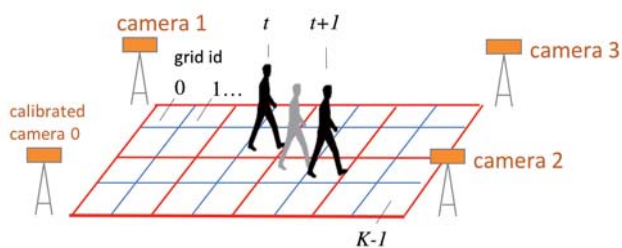


図 2: 追跡空間の分割

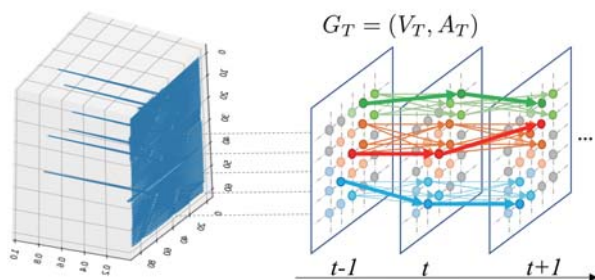


図 3: POM 及びネットワークの時間拡大

によって定まり, これを各フレームに対して時間拡大したネットワーク $G_T = (V_T, A_T)$ が⁸

$$V_T := \{1, \dots, T\} \times V$$

$$A_T := \{((t, i), (t+1, j)) \in V_T^2 \mid (i, j) \in A, t \in \{1, \dots, T-1\}\}$$

によって定まる. 以下では (t, i) から $(t+1, j)$ への枝 $((t, i), (t+1, j))$ を単に $e_{i,j}^t$ と表す. G_T の各点 (i, t) は時刻 t と物理的な位置 i の組によって定まっており, 各枝 $e_{i,j}^t$ は時刻 t において i から j へと 1 フレーム間で移動が可能であることを表している (図 3). あらかじめ求めてある, 時刻 $t \in \{1, \dots, T\}$ における領域 $i \in V$ 内の人物存在確率を ρ_i^t とする. このとき人物追跡の問題はグラフ G_T 上において尤度を最大化するような経路を出力する問題として定式化することができる.

2.2 整数計画問題としての定式化

定式化に際しては, 人流を G_T 上のフローと考えて定式化を行う. G_T 上のフローを表す変数 $f_{i,j}^t$ を

$$f_{i,j}^t = \begin{cases} 1 & \text{(時刻 } t, \text{ 領域 } i \text{ に人物が居り,} \\ & \text{その人物は次の時刻に領域 } j \text{ に移動)} \\ 0 & \text{(その他)} \end{cases}$$

となるように定める. このとき時刻 t において領域 i に存在する人数は $m_i^t := \sum_{j \in \mathcal{N}(i)} f_{i,j}^t$ である. したがって実行可能なフロー $\{f_{i,j}^t\}$ に対する対数尤度は

$$\begin{aligned} & \log \prod_{t,i} \left(\frac{\rho_i^{t(1-m_i^t)}}{(1-\rho_i^t)^{m_i^t}} \right) \\ &= \sum_{t,i} \log \left(\frac{(\rho_i^t)^{(1-m_i^t)}}{(1-\rho_i^t)^{m_i^t}} \right) \\ &= \sum_{t,i} (1-m_i^t) \log(1-\rho_i^t) + m_i^t \log \rho_i^t \\ &= \sum_{t,i} m_i^t \log \frac{\rho_i^t}{(1-\rho_i^t)} + \log(1-\rho_i^t) \end{aligned}$$

となる. したがって目的関数は

$$\sum_{t,i} m_i^t \log \frac{\rho_i^t}{1-\rho_i^t}$$

である.

制約にはフロー整合条件と各領域に存在する人数についての制約がある. 前者を単一起点・単一終点で表現するためにスーパーソースとスーパーシンクと呼ばれる仮想点 v_{source}, v_{sink} を追加する. 時間拡大ネットワーク上において, 人物が新しく出現する点は領域の端点及び $t=1$ に該当する点であるため, v_{source} からそのような点 (t, j) に向けて枝 $e_{v_{source},j}^t$ を張る. こうすることにより, フローの湧き出しを v_{source} のみであるとして問題を設定することができる. v_{sink} についても同様である. したがって制約は

$$\begin{aligned} f_{i,j}^t &\geq 0 && \forall t, i, j \\ \sum_{j \in \mathcal{N}(i)} f_{i,j}^t &\leq 1 && \forall t, i \\ \sum_{j \in \mathcal{N}(i)} f_{i,j}^t &\leq \sum_{k \in \mathcal{N}(i)} f_{k,i}^{t-1} && \forall t, i \\ \sum_{j \in \mathcal{N}(v_{source})} f_{v_{source},j}^t &\leq \sum_{j \in \mathcal{N}(v_{sink})} f_{j,v_{sink}}^t && \forall t, j \end{aligned}$$

となる。これらの制約によって、結果として得られるフローは v_{source} から v_{sink} への交わらない経路となり、これらを追跡結果とすることが可能となる。

[1] 内で、この問題内の制約は全ユニモジュラ行列として表されることが示されており、そのため上記の問題は変数を実変数に緩和することが可能である。また同論文内では上記の最適解を得るための方法として、作成したネットワーク上で枝の重みを変更しながら最短路を繰り返し解くことによって高速に最適解を得る方法について示されている。

3. KSP 適用範囲の分割方法

前述の方法による人物追跡の方法では、最短路の繰り返し計算による高速な計算手法が既に確立されているが、リアルタイムでの活用には至っておらず、即時的な活用には並列化などの工夫が求められる。特に対象となる領域が広い場合には、考えている制約上グラフの点数を増やす必要があり、最短路を解く際のグラフの点数が膨大となってしまう計算が遅くなる。一方、ネットワーク上での経路導出の際には 1 経路を出す毎に枝の重みを変更しながら計算を行っているため、導出に対しての直接の並列化は難しい。そこで本研究では上記のアルゴリズムを適用する時間を分割するための手法を提案する。

本研究では、追跡の難しさや追跡に成功したときの活用のし易さ等の観点から、主にスポーツを撮影した動画への適用を考えている。この際、特に球技のような 1 つの物体に対して多人数が集まるスポーツにおいては、領域の中での人の分布が偏ることが多い。人が集中している領域内では人物追跡やそれに用いる人物検出が難しく、人の目で見ても追跡がままならないような状況が生まれる一方で、その他の領域では人が少なく、追跡は非常に容易であると考えられる。このため映像内から混雑箇所の予想をすることができれば、混雑している箇所では検出誤差の補正を行うために局所的に KSP を適用し、これらの結果を繋ぎ合わせることで全体の追跡を復元することができる。

混雑の検知は各フレームごとに与えられる POM を用いており、各点に対してその周囲にある点における POM の和が一定以上である場合に混雑が起きているとみなす。例えば図 4 では、中央及びその左側の 2 点で混雑が検出される。また混雑検知結果は一般には連続的ではないため、混雑検知から一定時間以内に再度検知がなされた場合には混雑が継続しているとみなす。このしきい値のことを本研究内ではインターバルとよぶことにする。今回の KSP 分割に際しては、インターバルの設定により分割されている混雑の時間帯に対し、混雑が終了した時刻と次の混雑が発生した時刻の間で KSP を分割する(図 5 下部)。

4. 実験結果及び考察

本実験においては、Multi-camera Pedestrians Video データセット [4] の中からバスケットボールに関する映像を選択し実験に用いた。計算機環境を表 2、動画に関する情報を表 1 に示す。前ページで掲載してある画像の様に、動画内ではハーフコートでの競技の様子が撮影されており、混雑が頻発していることが見受けられる。人物検出に関しては YOLO [3] を、POM の生成及び KSP の適用についてはそれぞれ [2], [1] の実装を用いて計算を行った。

まず POM から得られた混雑検知結果を図 5 の上部で示す。横軸はフレーム数を表しており、図の上部に出ている番号では混雑が検知された時間を表している。また図の下部ではインターバルとして 3 秒を選択した場合、すなわち前後の混雑が 3

空間の広さ	15m × 15m
グリッド数	100 × 100
人数	13 人
KSP の深度	1
動画の長さ	360 秒
1 秒あたりフレーム数	25fps
カメラ数	4

表 1: 実験時の設定及び環境

CPU	Intel Xeon E5-2687W v4
アーキテクチャ名	Broadwell
CPU 周波数	3.00GHz
CPU 数	24(12 コア x 2 ソケット)
RAM	512GiB

表 2: 計算機環境

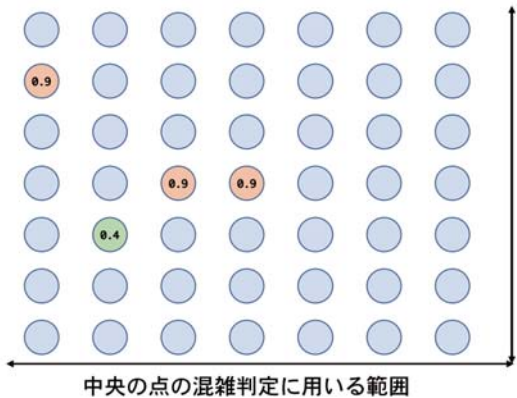


図 4: 各点に対する混雑判定に用いる範囲

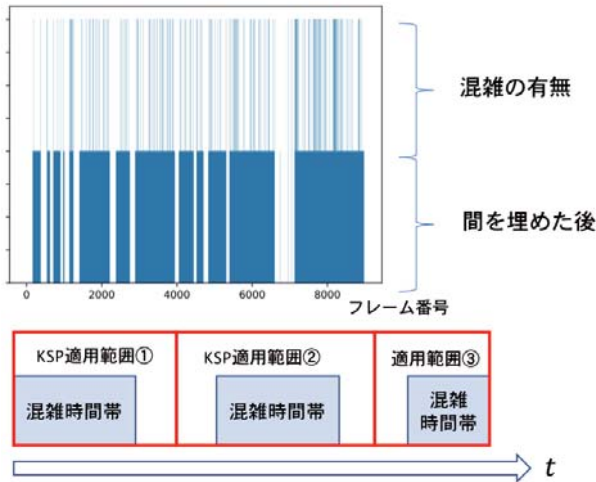


図 5: 混雑を基準とした KSP の分割方法

秒以内であった場合に混雑が継続していたとみなした場合の結果が示されており、上部と比較して連続的になっている事が分かる。インターバルの設定により混雑が発生している時間帯は16箇所分割されており、したがって KSP 適用範囲も16分割されている。この各区分において KSP を適用した結果及び分割前との比較結果を表3に示す。

まず計算時間については、各区分での計算時間は最大で164.62秒となり、分割前の KSP にかかっていた890.52秒を大きく短縮する結果となった。各区分での計算は全て独立に行うことができ、内部での並列化は行っていないため、全体で約5.4倍の高速化を実現することができる。

次に分割前と比較した精度であるが、追跡に際していずれかの失敗をしているフレームは約13.5%である1216フレームとなり、精度の低下はあまり見られない結果となった。

このうち位置を見誤るケースはグリッドの大きさが人間の大きさに比べ小さいため一概に誤りであるとは言えず評価が難しい。人数を見誤るケースについては、長時間他人と重なり続けたために起こる未検出が大きな原因となっている。未検出が起きている場合には、KSP の性質上未検出の前後で検出に成功している必要があり、この検出成功時間が短い場合には逆に成功した部分を過剰な検出とみなしてしまう。

例として5301フレーム目における図6の手前部分に未検出の人物が居るが、この人物は重なっている間しばらくの間未検出である。この部分については4768~5353フレームにて KSP の適用を行っており、この未検出が影響した結果5301フレームから5353フレームまでの約2秒間追跡が行えていない。このため分割に際しては未検出の取扱いが今後の主要な課題となる。特にスポーツの映像に関しては図6のように人が重なって起こる未検出が頻発するため、追跡を行うまでの全体の中で重なりに強い人物検出を行う必要がある。

5. まとめ・今後の課題

本研究においては、誤検出を考慮した動線追跡アルゴリズムである KSP を並列化するための分割手法を提案した。その結果、分割前と比べて9割以上の精度を保ちつつ5倍以上の高速化に成功した。今後の課題として、分割した際に前後の整合性を取るためのアルゴリズムの作成、未検出を考慮した分割方法の改善、空間に対する KSP の並列化が挙げられる。

1つ目については、今回行った KSP の分割に関して前後の人数や位置が合わないことが起こるため、全体で統一された人物追跡を行うために必要である。精度を損なわない分割ができている場合には分割の前後で最寄りの人物を同一とみなすなど単純な接続が可能であると見込まれるため、2つめに上げた分割手法の改善と合わせて研究を行っていく必要がある。

分割手法の改善に関しては、考察にて述べたように未検出が起こる区間をある程度含むような区間で分割する必要がある。映像であればカメラの位置、ビーコンであれば受信機の位置など、データ取得方法の特性に合わせて分割手法を考えることが望ましい。

最後に空間に対する並列化であるが、今回は混雑箇所に関わらず混雑が発生しているか否かのみによって分割を行っているため、混雑の発生地点を考慮していない。サッカーなど対象となる領域が広い場合には、KSP によって対処する混雑は全体の中でごく一部の狭い領域で発生するものであり、その際に全体を扱う必要はないと考えられる。混雑発生領域とそうでない領域を分けることで、精度を保ったまま更なる高速化が期待される。

分割前の計算時間	890.52 秒
分割後の最大計算時間	164.62 秒
全フレーム数	8999 フレーム
正解数	7783 フレーム
位置がずれた数	717 フレーム
人数がずれた数	499 フレーム
連続した未検知	3 箇所

表 3: 計算実験結果



図 6: 未検出の例

参考文献

- [1] Jérôme Berclaz, François Fleuret, Engin Türetken, and Pascal Fua, *Multiple Object Tracking using K-Shortest Paths Optimization*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 33, Issue 9, September 2011.
- [2] François Fleuret, Jérôme Berclaz, Richard Lengagne, and Pascal Fua, *Multi-Camera People Tracking With a Probabilistic Occupancy Map*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 2, pp. 267-282, February 2008.
- [3] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi, *You Only Look Once: Unified, Real-Time Object Detection*, CVPR 2016, OpenCV People's Choice Award, June 2016.
- [4] <http://cvlab.epfl.ch/cms/site/cvlab2/lang/en/data/pom>