

交差点の交通量調査におけるカメラ設置最適化の一提案

中野 勝章^{†1} 渡辺 魁^{†1} 中沢 実^{†1,†2,a)}

概要：近年、カメラ動画画像から画像認識技術を用いて道路交通量調査を行う研究が活発になっている。交差点におけるカメラ画像を用いた交通量調査では、単路環境に比べてカメラで捉えるべき範囲が広いこと、従来の固定型カメラ (CCTV カメラ) ではなく、可搬性を有する携帯型のカメラを用いることで交差点内の複数位置に設置することができる。それらを比較しカメラ設置位置を決めることで計測精度の向上が見込める。しかしながら、実際問題として交差点内に設置した携帯型カメラに対して、現地で最適なカメラ設置位置を評価する基準がなく、人手による経験的または直感的なカメラ配置となっている。そこで、本研究では短時間のカメラ動画画像から交差点における車両の流入口の位置関係に基づき、計測精度に影響を与える指標を算出し、それを利用することで最適なカメラ設置基準の判定を行っている。さらに、提案指標と OpenDataCam のカウンターライン機能を用いた計測精度との比較を実地検証を行い、提案した指標の妥当性を評価した。

キーワード： MOT (Multiple Object Tracking), Computer Vision, OpenDataCam, 交通量調査

A proposal for optimization of camera setting-up for traffic volume surveys at intersections

1. はじめに

全国道路・街路交通情勢調査とは、全国の道路における交通量等を把握し、道路の管理計画のための基礎資料を得る目的で国土交通省が5年に1度実施する調査である [1]。地方一般道における調査では人手による計測が主となっている [2] のが現状だが、国土交通省直轄の道路、特に単路の環境では CCTV カメラ画像を利用した AI 解析や常設トラフィックカウンタによる常時観測が実現されている [3]。国土交通省は、このような区間における人手による観測を令和3年度秋から廃止 [3] しており、センサや AI を活用した交通量調査の自動化への期待が高まっている。

しかし、今後想定される地方一般道における自動化を考慮すると、交差点のような複雑な環境で方向別での計測が要求される。特に交差点では単路に比べてカメラで捉えるべき範囲が広く、自治体によっては交差点全体を捉えられ

る程度のカメラ位置の高さを制約により確保できない場合がある。そのような状況では、従来の CCTV カメラのように一つの位置に固定すると、設置後の長時間計測の際に仮にイレギュラーな状況に直面した場合、撤去する際にコストがかかるという問題がある。したがって、交差点においては撤去の手間も少なく設置位置の自由度が比較的高い携帯型のカメラを活用することが必要となる。金沢市の支援を受け、金沢市内の交差点における携帯型のカメラを用いた実証実験 [4] では、交差点内の複数の箇所に柔軟にカメラ位置を決定できるというメリットがあった。一方で、そのように決定したカメラ位置における計測精度は動画画像の解析結果と人手による目視計測の結果を比較する必要がある。現地で設置したカメラ位置に対するフィードバックが即座に得られないという課題があった。

本研究は交差点内に設置した可搬型カメラで取得した短時間の画像データをもとに、最適なカメラ設置の評価を行うことを目的としている。物体追跡アルゴリズムによって動画画像から軌跡データを取得し、流入口の位置関係から計測精度に影響を与えるパラメータを推定し、その推定された指標によって最適なカメラ設置位置を決定する。さら

^{†1} 現在、金沢工業大学
Presently with Kanazawa Institute of Technology

^{†2} 現在、ロチェスター工科大学
Presently with Rochester Institute of Technology

a) nakazawa@infor.kanazawa-it.ac.jp

に、実装したシステムの妥当性を評価するために、金沢市内の3地点の交差点を対象に提案した指標と実際の計測精度との比較を行った。

本稿の構成は以下の通りである。第2章では関連研究について述べ、第3章ではカメラ位置の最適化手法について述べる。また、第4章では実装したシステムの妥当性の評価方法とその結果について説明し、最後に第5章で本研究のまとめと今後の課題について述べる。

2. 関連研究

2.1 MOT(Multiple Object Tracking)

MOTとは、動画内を移動していく複数の物体に対して、各フレームにおいてそれぞれに一意的IDを割り振ることで追跡を行うタスクである。追跡対象となるクラスは基本的に一つであるが、複数のクラスを同時に追跡するようなMOTモデルも存在する。MOTの手法はTracking-by-DetectionとOne-shotの二つに分類される。

Tracking-by-Detectionの手法は検出と追跡の2段階にタスクを分けて考える手法である。これによって、検出と追跡のそれぞれのタスクに対して適したモデルを選択できるというメリットがある。Tracking-by-Detectionの代表的な検出(Detection)モデルとしてYOLO[5]がある。この結果に対して、代表的な追跡(Tracking)モデルとしてSORT[6]やそれを拡張したDeepSORT[7]が挙げられる。一般的な処理の流れとしては、YOLOのような検出モデルによって推論された検出結果をSORTのような追跡アルゴリズムで処理することで追跡を実現している。一方で、One-shotの手法は検出と追跡をまとめて一つのモデルで行う手法で、一般的にTracking-by-Detectionよりも処理速度が速いという特徴がある。One-shotの代表的なモデルとして、FairMOT[8]やJDE[9]等が挙げられる。

また、このような物体追跡モデルの性能を評価する指標として、人物追跡ベンチマークのMOT challenge[10]が採用する20個の指標があり、その代表的な指標にMOTA(Multiple Object Tracking Accuracy)[11]がある。One-shotの手法はこのMOT challengeにパラメータが最適化されていることが多く、交通量調査では人や車両など対象とするクラスが多岐にわたり、One-shotの手法を拡張することが困難であるため、本研究で扱う手法はTracking-by-Detectionのみに限定する。

2.2 SORT(Simple Online and Realtime Tracking)

SORTとはTracking-by-Detectionにおいて代表的な追跡アルゴリズムであり、以下の2つの要素技術からなっている。

- (1) カルマンフィルタによる追跡物体の位置の事前予測
 - (2) 予測と実際の検出のIoUによる同一物体の判定
- カルマンフィルタによる予測について図1に示す。図1

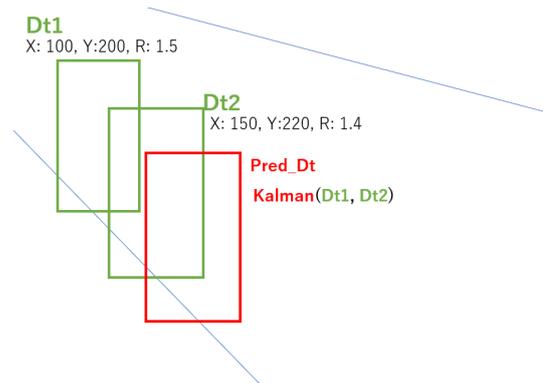


図1 カルマンフィルタによる予測

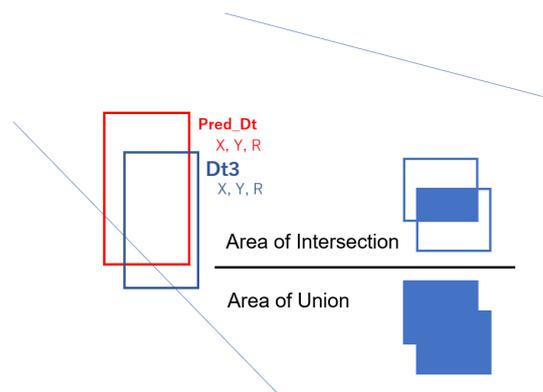


図2 IoUによるマッチング



図3 OpenDataCamのパスファインダー機能 [14]

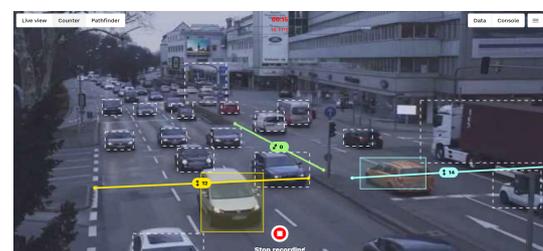


図4 OpenDataCamのカウンターライン機能 [15]

における緑色の矩形はYOLO等の検出モデルによって推論されたBounding Boxである。それぞれのBboxは中心座標(X,Y)とアスペクト比(R)の情報を持っている。このような状況でDt2という最新のBboxの状態とDt1という前の時刻におけるBboxの状態から次の時刻のBboxの状態の予測であるPred.Dtを出力する。

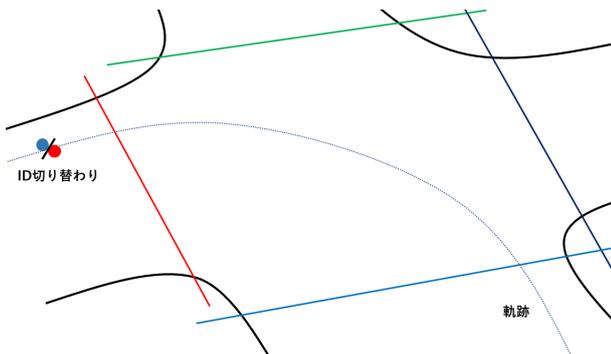


図 5 カウンターライン計測の一例

そして、図 2 のように次の時間に実際に検出された Dt3 と予測された Pred.Dt の重なりを IoU (Intersection over Union) という指標で評価する。この IoU がある閾値を超えていると Dt3 に Dt1 や Dt2 と同じ ID を割り振るアルゴリズムになっている。このような (1) と (2) の要素技術を組み合わせたアルゴリズムによって SORT による追跡が実現されている。

2.3 OpenDataCam

OpenDataCam[12] はリアルタイムで動画を解析することに特化したオープンソースツールである。主な機能として、パスファインダー機能とカウンターライン機能が GUI で提供されている。パスファインダー機能は、図 3 のように車や人などがどのように移動したかを軌跡として表示することができる。一方、カウンターライン機能は図 4 のようにユーザーが動画内に GUI を用いてラインを追加、削除することができる機能である。これによって、人や車などがラインを通過した数を計測できる。さらに、通過した物体の ID と時間まで記録可能なため、カウンターラインを複数設定することによって方向別で解析することが可能となる [13]。また、OpenDataCam はエッジコンピューティング向けに最適化されており、検出部を YOLO、追跡部は V-IOU tracker[16] というアルゴリズムを採用した Tracking-by-Detection の手法によって、リアルタイムな解析を可能にしている。

本研究では、提案指標と OpenDataCam のカウンターライン機能を用いた計測結果を比較することにより、提案指標の妥当性を評価する。

2.4 AI を用いた交通量計測サービス

近年、AI を利用した交通量計測を行う様々なサービスが登場している [17][18][19]。これらの中には、ユーザーがクラウドに解析対象の動画をアップロードして解析結果を得ることができるサービスもある。しかしながら、解析の適用範囲には制約があり、交差点のような複雑な環境において利用するには制限がかかる場合がある。単路環境に限れば、動画をアップロードする際に画角が解析に適切か

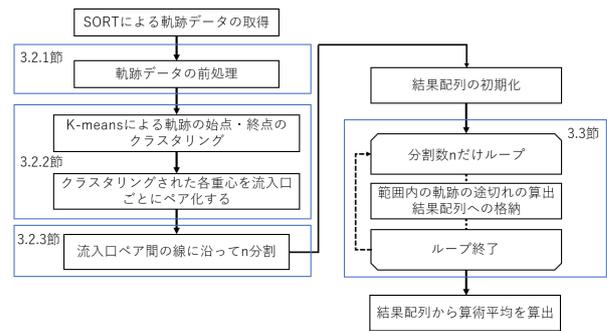


図 6 提案手法

どうかを判断する機能を持つサービスは存在するが、交差点環境においても設置した位置に対するフィードバックをユーザーに提供することは必要である。

3. 提案手法

3.1 概要

図 5 は交差点におけるカウンターライン計測の一例を表している。このような場合、仮にカウンターラインを引いた外側で ID が切り替わって消失した場合においても、計測精度に影響を及ぼさないことは明らかである。したがって、カウンターラインを用いた計測で重要となる範囲は、カウンターラインに囲まれる範囲である。提案手法では短時間の軌跡からこの範囲を推定し、その範囲内で途切れた軌跡の割合をカメラ設置位置の良さを表す指標とする。

図 6 に本方策に基づいた提案手法の流れを示す。具体的には、まず SORT で得た軌跡データに対して不要なデータを削除する前処理を行う。そして、軌跡の始点と終点それぞれについて K-means[20] によるクラスタリングを行う。その後、クラスタリングされた重心の位置関係に基づいて図 5 のような範囲の推定を行う。また、その範囲の大きさを変化させ、変化させた範囲ごとに途切れた軌跡の割合を算出し、それら全体の算術平均をカメラ設置位置の良さを示す指標とする。

3.2 範囲の決定

3.2.1 軌跡データの前処理

使用する軌跡データは図 7 のようになっているが、後のクラスタリング処理を考慮するとそのままのデータでは図 8 のように始点と終点が画像中央に見られ、クラスタリングが上手くいかないため十分な長さのない軌跡を排除する必要がある。そこで、前処理として始点と終点の直線距離が一定の閾値を上回っている軌跡のみを残す処理を行った。実装ではその閾値を 900 に設定しており、図 9 に閾値の基準とした長さを示す。図中の赤線が該当の距離であり、直交する軌跡の中で最短の軌跡の長さを採用している。

3.2.2 軌跡の始点・終点のクラスタリングとペア化

前処理を行った軌跡データに対して始点と終点それぞれ



図 7 軌跡データ



図 11 ペア化後の始点と終点の重心



図 8 始点 (赤), 終点 (緑) の生データ



図 12 流入口間の直線を 15 の点で分割した結果

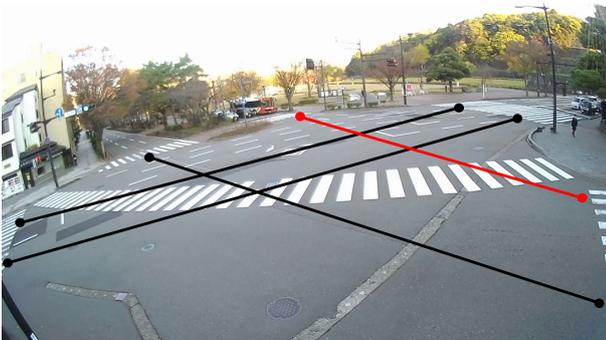


図 9 実装における閾値の基準となる長さ



(a) 決定範囲 1



図 10 前処理後のクラスタリング結果



(b) 決定範囲 2

図 13 範囲となる直線の例

表 1 広坂交差点 (十字路)

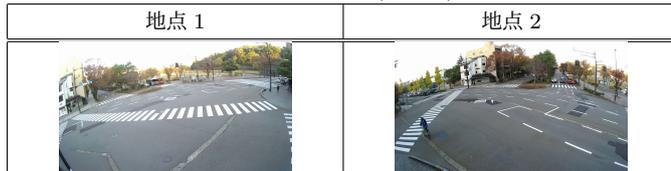


表 2 中四十万交差点 (十字路)



表 3 兼六坂上交差点 (三叉路)



に対して K-means アルゴリズムを用いたクラスタリングを行った結果を図 10 に示す。図 10 の交差点は十字路であるため、始点と終点はそれぞれ 4 クラスタにクラスタリングされる。また、図 10 において始点は赤の点、終点は緑の点で表され、黒い点は始点のクラスタの重心であり白い点は終点のクラスタの重心である。

そして、偏角ソート [21] を用いて同じ流入口の重心同士のパア化を行った結果が図 11 である。図 11 における黒い線で結ばれる 2 点は流入口を表し、白い線で結ばれる 2 点間は流入口の間を表す。

3.2.3 流入口間の線に沿った範囲の決定

次に、3.2.2 節で求めた流入口間の白線を n 個の点に分割する。実装上では 15 個の点に分割し、その結果を図 12 に示す。図中の水色の点は分割点を表し、図 14 のように各流入口のパアの線が中心に向かって縮小するように分割した点に沿って探索する。図 13 は実際に設定された直線であり、(a) の範囲から (b) の範囲に向かって分割された点を中心に向かって探索する。したがって、15 個の点に分割した場合 15 回範囲を探索し、これらの直線に囲まれる各範囲内で途切れた軌跡の割合を算出する。このように範囲を探索することによって、図 5 のようなカウンターラインを引き得る範囲を広く表現することを期待している。

3.3 指標の算出

指標の算出に利用するパラメータは、SORT アルゴリズムの中で一定期間での検出ともマッチングしなかった軌跡に割り振られる、軌跡の途切れを表すフラグである。

$$1 - (T_{break}/T_{all}) \quad (1)$$

式 1 は指標を表し、 T_{all} は決定範囲に少なくとも一点を持

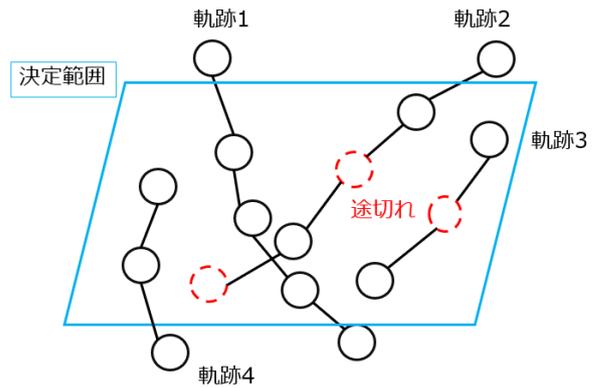


図 14 決定された範囲における指標算出の例

つすべての軌跡の数である。また、 T_{break} は決定範囲内に少なくとも一つの途切れフラグを持つ軌跡の割合である。よって、指標全体としては決定範囲を通る軌跡に対する一度も途切れていない軌跡の割合を表し、本指標は数値が高いほどカメラの設置位置における計測精度の見込みが高いことを示す。図 14 の場合、 $1 - (2/4) = 0.5$ と計算される。

4. 評価実験

4.1 実験方法

提案指標の妥当性を評価するために、金沢市内の 3 つの交差点内の 2 地点のカメラで撮影した動画 (信号 5 周期分、同時系列) に対して、提案指標と OpenDataCam のカウンターライン機能を用いた計測を行った時の計測精度を算出して比較を行った。表 1,2,3 は今回対象とした金沢市内の 3 つの交差点であるが、十字路と三叉路の両方について行うことで提案指標の拡張性を示す。比較対象とする OpenDataCam によるカウンターラインを用いた計測精度は式 2 で表される。

$$(gt - |gt - c|)/gt \cdot 100 \quad (2)$$

gt は目測の計測台数を表し、 c はカウンターラインでの計測台数を表す。各方向におけるこの計測精度を算出した平均値をカウンターラインの計測精度とする。

4.2 実験結果

表 4 広坂交差点 (十字路)

	地点 1	地点 2
指標	85.6%	79.4%
計測精度	79.2%	69.1%

表 5 中四十万交差点 (十字路)

	地点 1	地点 2
指標	90.4%	87.3%
計測精度	87.0%	76.5%

表 6 兼六坂上交差点 (三叉路)

	地点 1	地点 2
指標	90.1%	92.2%
計測精度	84.0%	91.6%

表 4,5,6 に評価結果を示す。十字路と三叉路のどちらの交差点に対しても、提案指標とカウンターラインの計測精度が比例の関係にあるため、評価結果は妥当であると言える。

5. おわりに

本稿では、交差点における可搬型カメラを用いた計測で設置したカメラに対して、短時間の軌跡データから計測精度に影響を与える指標を算出することで、最適なカメラ設置基準を提案した。提案した手法では、軌跡の始点と終点をクラスタリングした位置関係から、計測精度に影響を与える追跡の途切れの割合を算出し、カメラの設置を評価した。提案指標の妥当性を検証するため、金沢市内の 3 つの交差点に対してそれぞれの 2 地点からのカメラ動画画像から提案指標を算出し、OpenDataCam のカウンターラインを用いた計測精度との比較を行った。その結果、いずれの交差点についても提案指標とカウンターラインによる計測精度に比例の関係が見られ、実装した提案指標は妥当であるという結果が得られた。

今後の課題としては、今回の実験で使用した動画の長さは信号 5 周期分の長さだったため、より短い時間で同程度の指標が提示できるのかどうかの検証が必要であると考えている。また、交通量調査は数十時間に渡って行われるケースは珍しくないため、長時間の計測の中で起こる夜間撮影といったイレギュラーな状況にも対応できるようなシステムへの改良も検討している。

参考文献

- [1] 国土交通省：全国道路・街路交通情勢調査, https://www.mlit.go.jp/road/road_fr4_000071.html (参照 2022-04-02).
- [2] 国土交通省：平成 27 年度 全国道路・街路交通情勢調査一般交通量調査結果の概要について, <https://www.mlit.go.jp/common/001187536.pdf> (参照 2022-04-02).
- [3] 国土交通省：CCTV カメラ画像の AI 解析による常時観測に向けた精度検証, <https://www.mlit.go.jp/road/ir/ir-council/ict/pdf04/04.pdf> (参照 2022-04-02).
- [4] 金沢工業大学：情報工学科の中沢研究室が、日本海コンサルタントとともに、交差点における方向別交通量の AI 自動計測装置「携帯型トラモビ AI」の実証実験を実施。交通量の自動計測の社会実装に向けて、計測の精度向上などをめざす, https://www.kanazawa-it.ac.jp/kitnews/2021/1011_ai.html (参照 2022-04-02).
- [5] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi: *You Only Look Once: Unified, Real-Time Object Detection*, <https://arxiv.org/abs/1506.02640> (参照 2022-04-02).
- [6] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos,

- Ben Upcroft: *Simple Online and Realtime Tracking*, <https://arxiv.org/abs/1602.00763> (参照 2022-04-02).
- [7] Nicolai Wojke, Alex Bewley, Dietrich Paulus: *Simple Online and Realtime Tracking with a Deep Association Metric*, <https://arxiv.org/abs/1703.07402> (参照 2022-04-02).
 - [8] Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng, Wenyu Liu: *FairMOT: On the Fairness of Detection and Re-Identification in Multiple Object Tracking*, <https://arxiv.org/abs/2004.01888> (参照 2022-04-02).
 - [9] Zhongdao Wang, Liang Zheng, Yixuan Liu, Yali Li, Shengjin Wang: *Towards Real-Time Multi-Object Tracking*, <https://arxiv.org/abs/1909.12605> (参照 2022-04-02).
 - [10] MOTChallenge: The Multiple Object Tracking Benchmark, <https://motchallenge.net/> (参照 2022-04-02).
 - [11] Keni Bernardin, Rainer Stiefelhagen: *Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics*, <https://jivp-eurasipjournals.springeropen.com/articles/10.1155/2008/246309> (参照 2022-04-02).
 - [12] OpenDataCam 3.0.2 - An open source tool to quantify the world (GitHub), <https://github.com/opendatacam/opendatacam> (参照 2022-04-02).
 - [13] 渡辺 魁, 中野 勝章, 中沢 実: MOT を用いた交差点における交通量調査のカウンターラインの最適化手法の提案, 情報処理学会研究報告, 2021-DPS-189, Vol.2021, No.14, pp.1-6 (2021)
 - [14] OpenDataCam 2.0 - User Interface Walkthrough, <https://vimeo.com/346340651> (参照 2022-04-02).
 - [15] Using Nvidia Jetson and OpenDataCam to Explore Computer Vision and IoT Analytics, <https://www.hologram.io/blog/using-nvidia-jetson-and-opendatacam> (参照 2022-04-02).
 - [16] Python implementation of the IOU Tracker - bochinski - (GitHub), <https://github.com/bochinski/iou-tracker/> (参照 2022-04-02).
 - [17] AI 映像解析による交通量調査 - サーベイリサーチセンター, <https://www.surece.co.jp/solution/1567/> (参照 2022-04-02).
 - [18] 自動車交通量 - SmartCounter, <https://traffic-count.jp/vehicle-count/> (参照 2022-04-02).
 - [19] SCORER Traffic Counter, <https://www.scorer.jp/products/scorer-traffic-counter> (参照 2022-04-02).
 - [20] sklearn.cluster.KMeans scikit-learn 1.0.2 documentation, <https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html> (参照 2022-04-02).
 - [21] 多角形 (閉路) の作成～散在した点のソート～, <http://www5d.biglobe.ne.jp/~tomoya03/shtml/algorithm/Heiro.htm> (参照 2022-04-02).