G-19

深層学習による物体検出を用いた交通量計測

Traffic Volume Measurement using Object Detection by Deep Learning

○斎藤大嗣[1]，　関弘翔[2]，　細野裕行[2]
*Daishi Saito[1], Hiroto Seki[1], Hiroyuki Hosono[2]

Abstract: A traffic jam in developing countries has become serious and has developed into various social problems. The purpose of this study is to measure traffic volume with high accuracy. This information is necessary to reduce traffic volume. In this paper, we studied the method of detecting vehicles by deep learning, tracking vehicles based on the results, and measuring traffic volume using tripwire.

１．まえがき

新興国における交通渋滞は深刻化しており，環境破壊，経済損失といった問題につながるため解決すべき重要な課題である．交通渋滞を解消するためには現在の交通状況を知る必要があるが，新興国における交通量計測には課題が多い．

例えばインドでは，特にバイクが多く走行しており，またオートリクシャーと呼ばれる 3 輪車もあるため，それらの小さい車両が密集して 1 車線で横に複数台並ぶような混雑した交通状況になっている．さらに，このように混雑することで物体が複雑に重なり遮る（occlusion）問題も発生する．現在インドには，画像式車両検知器が既設されているが，車線毎に交通量の計測を行うものであり，車長などの簡単な特徴で車種を判別している．この手法では，車両が重なっていたり，小さかったりすると重複検知や誤判別により計測精度が悪くなる問題がある．

そこで本研究は，インドのように 2 輪車等が多く混雑した交通状況でも交通量の計測を正確に行うことを目的に，車線に関係なく車両を検出・判別可能な深層学習による物体検出を用いた交通量計測を検討する．

具体的には，インドの道路を走行している車両（オートリクシャー，二輪車，バス，車，トラック）の種類ごとに物体検出できるようにモデルに学習させ，その検出結果を基に交通量の計測を行う．最終的な交通量計測の精度 80%以上を目標とする．

２．物体検出を用いた交通量計測

交通量計測の方法としては，まずカメラから取得した映像ファイルから 1 フレーム取り出す．次に，対象とする車両を検出できるように学習した物体検出モデル（本報告では YOLOv3[1]）を用いて，1 フレームに対する車両検出結果を得る．数フレーム間の車両検出結果を用いて，重なり具合を示す IoU（Intersection over Union）に基づいた車両追跡を行う．追跡車両が，画像中に設置するトリップワイヤを通過したか否かで車種ごとに台数を計測する．

２.１．　YOLOv3 による物体検出

YOLO（You Only Look Once）は CNN（Convolutional Neural Network）を用いて物体検出を行うモデルである．CNN を用いた物体検出の代表例として Faster-RCNN が挙げられる．これは候補領域検出を行う RPN（Region Proposal Network）と呼ばれる CNN と，画像分類を行う CNN の 2 つのネットワークから構成され，入力画像に RPN に入力して得られた候補領域をそれぞれ CNN に入力して結果を得る方法である．この方法では候補領域が多い場合，それぞれに対して CNN に通す必要があるので，膨大な計算時間を要し，リアルタイム物体検出には向いていない．一方で，YOLO は一枚の画像を一度だけ CNN に通して処理を行うため計算時間が少なくて済む．

YOLOv3 では特徴抽出器として DarkNet-53 と呼ぶ画像識別用の CNN を提案しており，識別用に学習した DarkNet-53 からクラス分類を行う全結合層を省き用いている．DarkNet-53 から 3 サイズ（小・中・大）に相当する大きさの特徴マップを取り出して各特徴マップの各ピクセルにアンカーボックスと呼ばれる検出候補を 3 つ用意し，最終的に各サイズで場合分けして Convolution を行うことで検出結果であるバウンディングボックスの座標，物体らしさのスコアと各クラスの存在確率を算出する．

本報告では，Github 上[2]に公開されている，COCO データセット[3]を学習した YOLOv3 の重みを初期値として転移学習を行い，20 エポックで学習した．学習に用いたデータの詳細を Table 1 に示す．このデータは，物体検出用のデータセットを含む COCO と Pascal VOC[4]

１：日大理工・学部・情報　２：日大理工・教員・情報

から，必要なクラス（自転車，バイク，バス，車，バイク，トラック）のデータのみ取り出して用意しており，オートリクシャーについてはインドの交差点映像からアノテーションし独自にデータを作成した．

Table 1.　Dataset

class name	number of objects
auto-rickshaw	1946
two-wheeled vehicle	12658
bus	5695
car	31278
truck	7620

2.2.　IoU トラッキング

論文[5]を参考に，複数フレームの車両検出結果からIoU を用いて車両を追跡する．IoU とは式(1)で示すように，2 つのバウンディングボックス間でどれだけが重なっているかを 0~1 の数値で求める指標である．IoU トラッキングの流れとしては，現フレームと 1 フレーム前それぞれの検出結果を比較して一番高い IoU を持つものを見つけ，その IoU の値が設定した閾値以上なら，同一車両として追跡する．

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \qquad (1)$$

2.3.　トリップワイヤによる交通量計測

画像中の任意の位置にトリップワイヤ（Fig. 1）を設置し，追跡中の車両のバウンディングボックスの中心座標がトリップワイヤ付近にあればクラスごとに台数を計測する．具体的には，バウンディングボックスの中心座標とトリップワイヤの両端の点との長さの和（$d_1 + d_2$）と,トリップワイヤの長さ（d_3）との差を求めて，差が閾値以内なら 1 台通過としてカウントする．

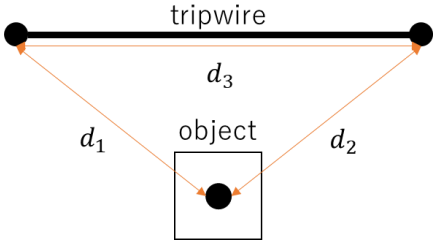

Figure 1.　Tripwire

3．実験結果

本報告では，インド・アーメダバード市にあるパルディ交差点の約 30 分間の動画に対して交通量を計測した．目視による計測結果と本システムによる計測結果の比較を Table 2 に示す．結果からトラック以外は89%以上の精度で計測できている．トラックの認識精度が悪いのは，画像中の物体が小さいとリクシャーがトラックとして認識されることが原因である．.全車両での結果（all）を見ると，未検出は少なく，余分に検出している．これは，学習時の画像中に存在する物体よりも，対象の物体が小さいため,安定した検出ができず,多重検出したことが原因である．

Table 2.　Comparison of measurement results

	auto-rickshaw	two-wheeled	bus	car	truck	all
ground truth	575	976	90	340	6	1987
predict	559	975	80	321	181	2116
error ratio[%]	2.78	0.10	11.1	5.59	2920	6.49

4．まとめ

本報告では，YOLOv3 による検出結果から IoU トラッキングを行い，交通量を計測した．目視の計測結果との比較から，オートリクシャーがトラックとして誤判別される問題があることが分かった．計測精度向上のためには，小さい物体でも安定して検出できるモデルに変更した上で，オートリクシャーのデータセットを増やして誤判別をできるだけ無くす必要がある．

5．謝辞

本研究は,JST と JICA 共同の研究プログラム SATREPS (ID:16667566)の一環で行われたものです。

6．参考文献

[1] J. Redmon and A. Farhadi: “YOLOv3: An Incremental Improvement,” arXiv:1804.02767 (2018)
[2] GitHub:「tensorflow-YOLOv3」, https://github.com/YunYang1994/Tensorflow-YOLOv3（2019-09）
[3] T. Lin; et al.: “2014. Microsoft COCO: common objects in context,” In ECCV 2014, pp.740-755 (2014)
[4] M. Everingham, et al.: “The Pascal Visual Object Classes (VOC) Challenge,” International Journal of Computer Vision, vol. 88(2), pp. 303-338 (2010)
[5] E. Bochinski, V. Eiselein, T. Sikora: “High-Speed Tracking-by-Detection Without Using Image Information,” IEEE Int. Conf. on Advanced Video and Signal Based Surveillance (AVSS) (2017)