FlowNetC を導入した D&T による物体検出の高精度化

瀬尾 俊貴 †　福井 宏 †　平川 翼 †　山下 隆義 †　藤吉 弘亘 †
† 中部大学
E-mail: seotoshiki@mprg.cs.chubu.ac.jp

1　背景・目的

物体検出は，画像中の物体の位置と大きさを推定する技術であり，自動車の運転支援システムや監視カメラなどに幅広く応用されている．CNN を用いた従来の物体検出手法 [1][2][3][4] では，単一フレームを対象とした処理を前提としており，1 フレーム中に含まれる物体位置を正しく推定するようにネットワークを学習する．一方で，複数フレームでの動きを考慮することで，あるフレームにおいてオクルージョンが発生する場合でも，前後の関係性および追跡から物体を検出でき，検出精度の向上が期待できる．

動きを考慮した物体検出の高精度化を実現するために，本稿では Detect to Track and Track to Detect (D&T) [5] をベースに，動き情報であるオプティカルフローに着目する．D&T では，物体検出と追跡を同時に行うことで物体検出を高精度化している．D&T は，各フレームで物体検出を行い，前後のフレームにおける移動方向と移動量を，相関層と呼ぶ前後のフレームの変位を獲得する隠れ層から推定している．しかし，この移動方向と移動量の推定では，画像全体または局所的な画素の変化を捉えず，検出領域の類似性から移動量を推定している．そのため，これらの推定が正しく出来ないことがある．また，オプティカルフローを Fully Convolutional Neural Network [6] を用いて推定する手法に FlowNet [7] が提案されている．FlowNet は，前後のフレーム画像を結合した画像を入力とする FlowNetS，前後のフレーム画像を別々に入力し，相関層により変位を獲得する FlowNetC から構成される．

本来，D&T から出力されるバウンディングボックスの移動方向と，オプティカルフローの方向は一致すべきである．そこで，本研究では D&T の前後のフレームにおける物体候補領域が移動すべき方向を決定するために，D&T から出力される移動量とオプティカルフローの角度誤差関数 L_{rad} を追加し，物体検出精度を向上させることを目的とする．また，D&T と FlowNetC のマルチタスク化を行うことで精度の向上ができるかを検討する．評価実験では，従来手法，D&T と FlowNetC のマルチタスクモデル，マルチタスクモデルに角度誤差関数 L_{rad} を追加した手法のオプティカルフローと物体検出精度の定量的，定性的な評価を行う．

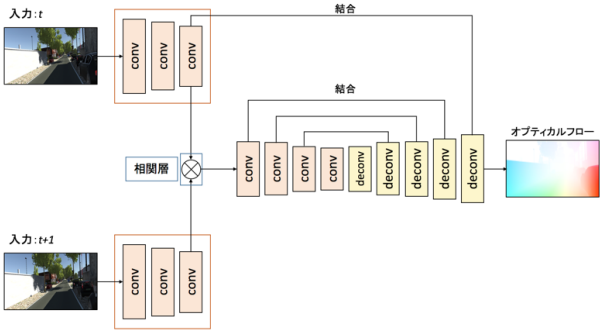

図 1　FlowNetC のネットワーク構造

2　関連手法

特徴マップ間の相関をオプティカルフローを求める手法として FlowNet と物体検出と追跡に用いる Detect to Track and Track to Detect (D&T) が提案されている．以下にこの 2 つの手法について説明する．

2.1　FlowNet

Fully Convolutional Neural Network(FCN) を用いてオプティカルフローを推定する手法として FlowNet [7] が提案されている．FlowNet には，前後のフレーム画像を結合した画像を入力として畳み込み処理と逆畳み込み処理を行う FlowNetS と前後のフレーム画像を別々に入力し，相関層から得られた変位をもとに逆畳み込み処理を行う FlowNetC から構成される．提案手法に用いる FlowNetC のネットワーク構造を図 1 に示す．

ここで，相関層では前後のフレームの入力画像に対して畳み込み処理を複数回行い，得られた特徴マップ $f_1(x_1)$, $f_2(x_2)$ から，移動幅により一致した座標の特徴マップ $f_1(x_1+o)$, $f_2(x_2+o)$ を式 (1) により 1 つの特徴マップに結合することで，移動量の推定に有効な変位を獲得できる．各移動幅に応じた大〜小変位の特徴マップが生成され，物体の移動量推定に用いる．

$$c(x_1, x_2) = \sum_{o \in [-k,k] \times [-k,k]} \langle f_1(x_1+o), f_2(x_2+o) \rangle \quad (1)$$

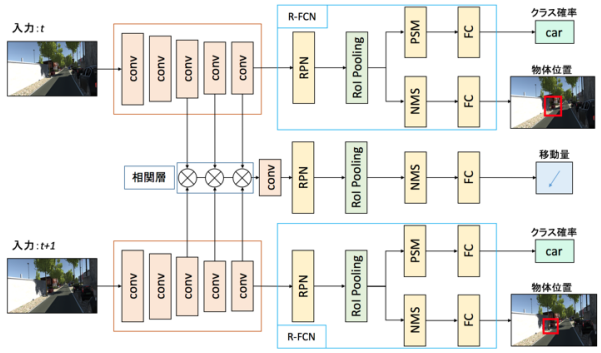

図 2　D&T のネットワーク

ただし，x_1，x_2 はフレーム間の各座標，k は x_1, x_2 を中心とした注目領域，o は移動幅である．

2.2　Detect to Track and Track to Detect

動画を対象に物体検出と追跡を同時に行う手法として Detect to Track and Track to Detect (D&T) が提案されている．D&T は Object Detection via Region-based Fully Convolutional Networks (R-FCN) [8] 内部の Region Proposal Network [9] で検出された物体候補領域の相関特徴マップから各物体の移動量と移動方向を推定し，各物体の相互関係から物体検出と追跡を行う．D&T のネットワーク構造を図 2 に示す．

D&T の物体検出は，R-FCN から各フレームの物体検出を行うブロック，移動量を推定する相関層と呼ぶ隠れ層から次のフレームの移動量を推定し物体候補領域を決定するブロック，相関層から得られた物体検出結果と次のフレームの R-FCN による物体検出結果の Intersection over Union (IoU) から最終的な物体らしさを決定するブロックの計 3 つから構成されている．

得られた相関特徴マップは，畳み込み処理を行った後，RPN へ入力，物体候補領域検出及び移動量の推定に用いる．その後，相関層から得られた物体検出結果と次のフレームの R-FCN による全ての物体検出結果をノードとして結合し，対応付けを行う．この時，IoU が 0.5 以上の物体候補領域に最終的な物体らしさを加算することにより R-FCN を単体で用いるよりも高精度な物体検出を実現している．しかし，D&T の相関層から推定する移動量は，相関層から出力された特徴マップからそれぞれの検出領域の IoU を算出し移動量を推定している．画像全体または局所的な画素の変化を捉えていないため，移動量推定が正しく出来ないことがある．

3　提案手法

本研究では，D&T 及び FlowNetC の両手法に用いられている相関層に着目し，マルチタスク化を行う．また，前後のフレームにおいて D&T が決定すべき次のフレームでの移動方向をオプティカルフローに着目した

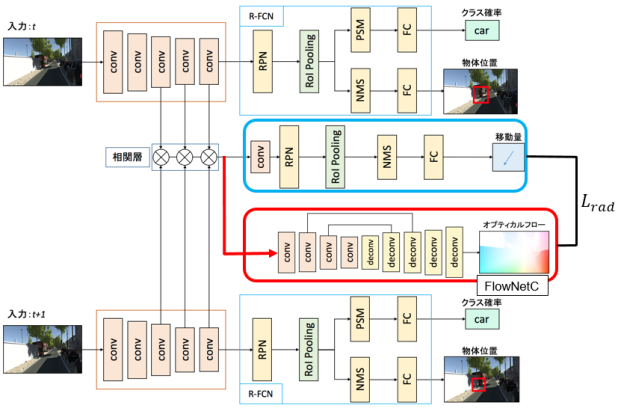

図 3　提案手法のネットワーク

損失関数 L_{rad} を用いて獲得できるようにする．これにより，移動方向を適切に決定する学習が可能となる．提案手法は大きく分けて 2 つの要素から構成される．提案手法のネットワークでは，はじめに R-FCN で前後のフレームの物体検出を行う．その後，3 から 5 回の畳み込み処理で得られた特徴マップをそれぞれ相関層へ入力し，相関特徴マップを生成する．そして，生成された相関特徴マップを，物体候補領域を求めてバウンディングボックスの移動方向と移動量を推定するブロックと，畳み込み処理と逆畳み込み処理によりオプティカルフローを推定するブロックの 2 つへ入力する．そして，物体検出とオプティカルフローを同時に出力する．以下に，各処理について詳細に述べる．

3.1　マルチタスクラーニングの導入

D&T と FlowNetC は，フレーム間の変位を求める処理が共通している．両手法では共に相関層を用いる点においても類似しており，マルチタスク化することで両タスクの精度向上を図ることが可能である．また，D&T は物体ごとに変位を算出するが，その際，FlowNetC による画素ごとのフロー情報は，補助情報として物体検出に有効であると考えられる．そこで，図 3 のように相関層を共通化し，物体検出結果とオプティカルフローを同時に出力するネットワークを提案する．

3.2　角度誤差関数の定義

提案するマルチタスクラーニングのネットワークでは，式 (2) で示す損失関数の総和を用いて学習を行う．ここで，L_{class}(クロスエントロピー誤差関数) は R-FCN のクラス識別スコアの損失，L_{loc}(smooth L1 誤差関数) はバウンディングボックスの損失，L_{flow}(平均二乗誤差関数) はオプティカルフローの損失，L_{tra}(smooth L1 誤差関数) は物体追跡の損失である．

$$L_{all} = L_{class} + L_{loc} + L_{flow} + L_{tra} \tag{2}$$

ここに，D&T で推定する移動量 $M_{(x,y)}$ と物体候補

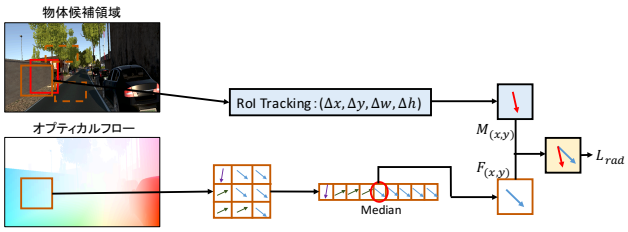

図 4　角度誤差関数の決定方法

領域のオプティカルフローの真値 $F_{(x,y)}$ の差異を吸収する目的で以下の式 (3) で示す損失関数を追加する．物体候補領域のオプティカルフローは，図 4 に示すように全ての画素のオプティカルフローをもとに勾配ヒストグラムを作成し，最も近い角度が含まれているオプティカルフローを教師信号として逆伝播する．

$$L_{rad} = \begin{cases} (M_{(x,y)} - F_{(x,y)})^2 & if(|M_{(x,y)} - F_{(x,y)}| < \pi/4) \\ 0 & (otherwise) \end{cases} \tag{3}$$

4　評価実験

本手法の有効性を示すため，D&T と提案手法の定量的な精度の比較を行う．以下にデータセット，学習条件，及び実験結果の詳細について述べる．

4.1　データセット

本実験では Virtual KITTI [10] データセットを用いて実験を行う．Virtual KITTI データセットは 3D モデルで作成された一般車道を走行するデータセットである．また，rain，fog，overcast といった天候や sunset，noon といった時間帯，同フレームにおける見え方を考慮した動画像も別々に含まれているだけでなく，2D，3D のマルチオブジェクトトラッキングやカテゴリ，オプティカルフロー，深度の真値も用意されている．本実験では学習に 1,014 枚，評価に 225 枚を用いて，自動車の物体検出とオプティカルフローの推定を行う．

4.2　評価指標

物体検出の精度指標には物体検出タスクにおいて使用される mean Average Precision(mAP) を用いる．mAP とは，全評価画像に対する推定されたバウンディングボックスの内，実際に真値が存在した割合の平均値を表す．また，オプティカルフローの精度指標には Average EndPoint Error(AEPE) を用いる．AEPE とは画像サイズ $m \times n$ の全評価画像 N に対する画素ごとの真値ベクトル (x, y) と推定値ベクトル (x', y') の平均二乗誤差を用いて式 (4) のように表す．最後に，物体をどれだけ追跡できたかを示す指標として，物体追跡率 $Track_{acc}$ を定義する．物体追跡率はフレーム $t, t+1$ における両フレームに存在するバウンディングボックスの真値の総数 $gtBox_{(t)} \cap gtBox_{t+1}$ の内，フレーム $t, t+1$ のバウンディングボックス $Box_{(t)}$ および Box_{t+1} においてどれだけ物体を検出できたかを示す割合であり，式 (5) のように表す．

$$AEPE = \frac{1}{N}\sum_{i=0}^{m}\sum_{j=0}^{n}\{(x_{(i,j)} - x'_{(i,j)})^2 + (y_{(i,j)} - y'_{(i,j)})^2\} \tag{4}$$

$$Track_{acc} = \sum_{i=0}^{t}\frac{Box_{(t)} \cap Box_{(t+1)}}{gtBox_{(t)} \cap gtBox_{t+1}} \tag{5}$$

4.3　実験結果

従来手法には D&T，FlowNetC を用いる．また，提案手法は D&T と FlowNetC をマルチタスクとして学習するモデル (D&T+Flow)，さらに角度誤差関数 L_{rad} を追加して学習するモデル (D&T+Flow+L_{rad}) である．各結果を表 1 に示す．マルチタスク化した D&T+Flow は従来手法である D&T と比べ，mAP が 0.4%低下した．一方，D&T+Flow+L_{rad} により適切なバウンディングボックスを選択する学習を行ったことで，従来手法と比べ mAP が 0.9%向上し，AEPE も 0.5pixel 削減できた．さらに，$Track_{acc}$ に着目すると従来手法と比べ 12.2%精度が向上したことが確認できる．これは，D&T と FlowNetC のマルチタスク化による効果的な物体の移動方向の予測による結果だと言える．以上より，提案した損失関数 L_{rad} の有効性が確認できる．

表 1　各モデルの精度比較

Method	mAP [%]	AEPE [pixel]	$Track_{acc}$[%]
D&T	86.4	-	71.2
FlowNetC	-	22.1	-
D&T+Flow	86.2	23.0	82.9
D&T+Flow+L_{rad}	**87.3**	**22.5**	**83.4**

また，マルチタスクラーニングモデルにおける物体検出とオプティカルフロー出力例と数フレームにおける物体検出例を図 5，6 に示す．ここで，赤い矩形が真値，青い矩形が推定値である．図 5 を見ると，角度誤差関数 L_{rad} の導入により一般車の物体検出が位置によらず検出できていることが確認できる．また，移動する自動車のオプティカルフローも同時に推定できていることが確認できる．さらに，図 6 では D&T+Flow では検出に失敗した画像中心あたりの自動車を，D&T+Flow+L_{rad} は向きが一致するように学習でき，連続したフレームで自動車の検出が行えるようになった．

5　おわりに

本研究では, 物体検出である D&T とオプティカルフローの推定手法である FlowNetC のマルチタスク化を

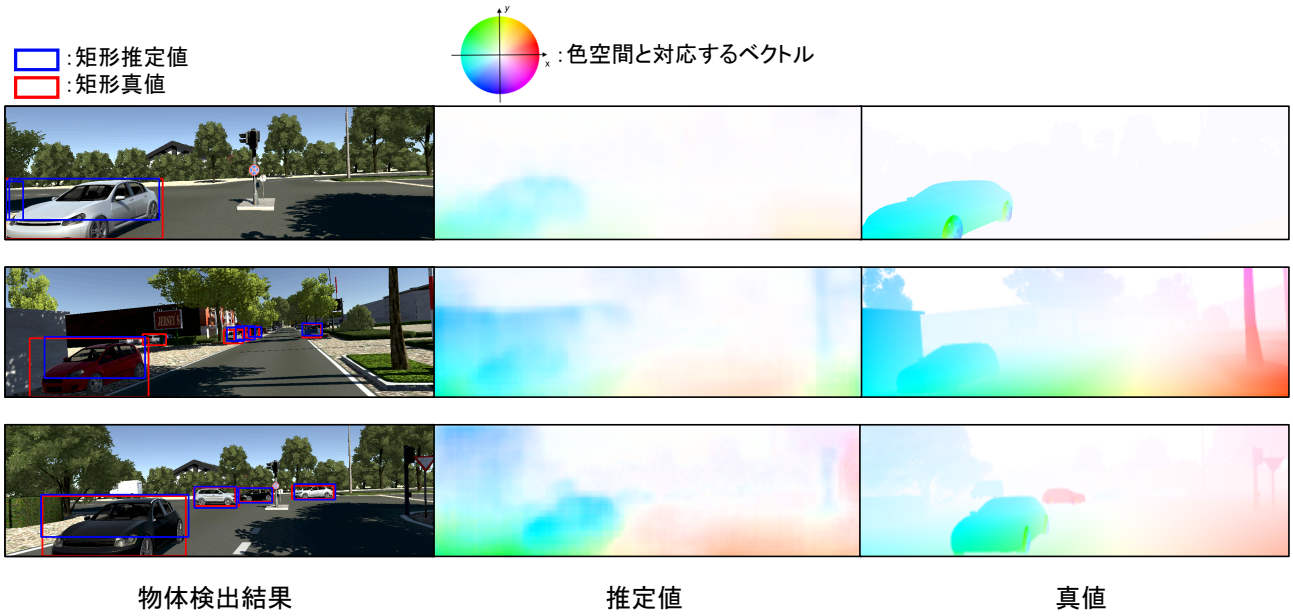

図 5　出力結果

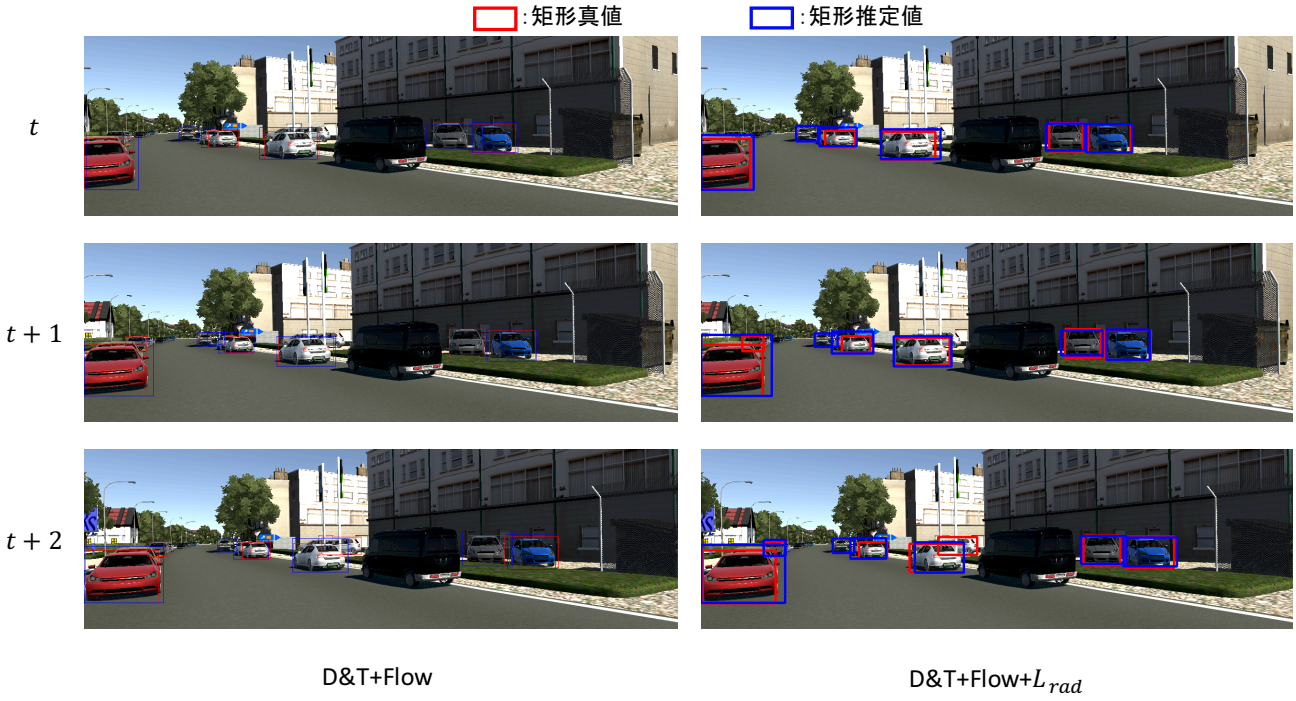

図 6　数フレームにおける物体検出例

行い学習，推定する手法を提案し，学習時にオプティカルフローを考慮した角度損失関数 L_{rad} を定義することで物体検出精度，及び物体追跡精度を向上させた．今後の課題として，評価時にオプティカルフローを考慮した時の精度向上や学習コストの削減を目指す．

6　謝辞

本研究（の一部）は，総合科学技術・イノベーション会議の SIP（戦略的イノベーション創造プログラム）「自動運転（システムとサービスの拡張）のうち自動運転技術（レベル 3、4）に必要な認識技術等に関する研究」（管理法人：NEDO）によって実施されました．

参考文献

[1] Redmon, Joseph, et al. ”You only look once: Unified, real-time object detection.” Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[2] J. Redmon and A. Farhadi. YOLO9000: Better, Faster, Stronger. Computer Vision and Pattern Recognition, 2017.

[3] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016, October). Ssd: Single shot multibox detector. In European conference on computer vision (pp. 21-37). Springer, Cham.

[4] Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation", Computer Vision and Pattern Recognition, 2014.

[5] Christoph. Feichtenhofer, Axel. Pinz, Andrew. Zisserman, "Detect to Track and Track to Detect", International Conference on Computer Vision, 2017.

[6] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in Computer Vision and Pattern Recognition, pp. 3431-3440, 2015.

[7] Philipp Fischer, Alexey Dosovitskiy, Eddy Ilg, Philip Husser, Caner Hazrba, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers, Thomas Brox, "FlowNet: Learning Optical Flow with Convolutional Networks", Computer Vision, 2015.

[8] Jifeng Dai, Yi Li, Kaiming He, Jian Sun, "R-FCN : Object Detection via Region-based Fully Convolutional Networks", in Neural Information Processing Systems, 2016.

[9] Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", Computer Vision and Pattern Recognition, 2015.

[10] Gaidon, A and Wang, Q and Cabon, Y and Vig, E, "Gaidon:Virtual:CVPR2016", Computer Vision and Processing Systems, 2016.