1Q-04

Vision Transformer を用いた物体追跡モデルのパラメータ探索についての一研究

福嶋大樹　　石川知一
東洋大学

1. はじめに

近年、コンピュータの著しい性能向上と記憶媒体の大容量化により、かつては難しかった実時間処理が可能になった事に加え、動画像を蓄えられるようになり、機械学習による動画像処理の研究が盛んになった。

動画像処理の研究の一例として、物体検出タスクの技術を応用させた研究が物体追跡タスクである。物体追跡タスクは与えられた動画像から、指定した対象がどのように移動するかを推定する問題である。

2020 年には Dosovitskiy らの研究で Neural Machine Translation の分野で利用されていた Transformer を画像分類タスクに適用した Vision Transformer(以下、ViT と略す)が開発された。しかし、物体追跡タスクで Transformer を適用した例は少なく、現在提案されている物体追跡タスクの手法を改良することによって精度を向上させられるのではないかと考えた。

物体追跡タスクの応用例として、動画内の移動物体に対してのオートフォーカスや、動画像内の追跡対象物に対してバウンディングボックスで強調表示をする編集を行うことで、スポーツのリプレイ映像で選手を強調表示させられることができる。

2. 関連研究

従来の物体追跡タスクでは、シャムネットワークにもとづく手法が注目されていたが、シャムネットワークの主な問題は、背景領域や過去の追跡フレームからの情報をモデル予測に組み込むことができない点であり、Bhat らはシャムネットワークの問題点を解決するための新しいトラッキングアーキテクチャとして Discriminative Model Prediction (以下、DiMP と略す)を提案した [1]。DiMP はシャムネットワークモデルとは異なり、追跡物体と背景情報を完全に利用することで、物体への識別力を失うことなく、数回の学習でターゲットモデルの予測を可能にしている。しかし、DiMP はテンプレート画像とサーチ画像から抽出された特徴量が相互相関の演算に利用されるため動画像全体の文脈は利用されていない。

Zhao らはシャムネットワークに ViT を組み合わせ、自己注意および交差注意構造を用いることで、動画像全体の文脈を利用して物体追跡を可能とした Tracker with Transformer(以下、TrTr と略す)を提案した [2]。本研究では Zhao らが提案した TrTr のモデルを使用する。

3. 実験方法

本研究の目的は物体追跡の精度向上で、本実験では損失関数のパラメータを調整する実験を行い、精度向上を図る。

TrTr では Transformer モジュールの出力後に 3 つの独立したヘッドが接続されており、1 つは対象物の分類用としてマップYを生成し、残りの 2 つは回帰用としてマップOとマップSを生成し、物体の位置特定を行う。

マップYは出力ストライドによって離散化された低解像度におけるサーチ画像中の対象物の出現確率に相当する。ここで、出力ストライドによる離散化誤差を減らすために、局所的なオフセットをマップOとして追加で予測する必要があり、バウンディングボックスサイズ回帰のためにマップSを予測する必要がある。

予測されたマップYと比較を行うためにサーチ画像中の対象物の中心の正解の値$\overline{p}$に対して、低解像度において$\tilde{p} = \left(\left\lfloor\frac{\overline{p}_x}{s}\right\rfloor, \left\lfloor\frac{\overline{p}_y}{s}\right\rfloor\right)$を計算する。ここで$s$は出力ストライドである。次に、ガウスカーネル$\overline{Y} = \exp\left(-\frac{(x-\tilde{p_x})^2+(y-\tilde{p_y})^2}{2\sigma_p^2}\right)$を用いて、予測されたマップ$Y$と比較する。ここでガウスカーネルに含まれる$\sigma_p$はオブジェクトサイズに適応した標準偏差である。

$$L_Y = -\sum_{xy}\begin{cases}(1-Y_{xy})^{\alpha}\log(Y_{xy}) & (\bar{Y}_{xy}=1)\\ (1-\bar{Y}_{xy})^{\beta}(Y_{xy})^{\alpha}\log(1-Y_{xy}) & (otherwise)\end{cases} \quad (1)$$

ここで、Y_{xy}は(x, y)におけるマップYの値である。αとβは先行研究 [3]に従って$\alpha = 2$と$\beta = 4$を使用している。次に、オフセット回帰の損失関数はL^1損失を用いて以下の式としている。

$$L_O = \left|O_{\tilde{p}} - \left(\frac{p}{s} - \tilde{p}\right)\right| \quad (2)$$

ここで、$O_{\tilde{p}}$は$\tilde{p}$におけるマップ値である。正解のバウンディングボックスのサイズ$(\overline{w}_{bb}, \overline{h}_{bb})$の損失関数に対しても、$L^1$損失を用いて以下の式を用いている。

$$L_S = \left|S_{\tilde{p}} - \tilde{s}\right| \quad (3)$$

ここで、$S_{\tilde{p}}$は$\tilde{p}$におけるSのマップ値である。また、$\tilde{s} = \left(\frac{\overline{w}_{bb}}{W}, \frac{\overline{h}_{bb}}{H}\right)$は正規化された正解のバウンディングボックスで

A Study on Parameter Search of Object Tracking Model Using Vision Transformer
DAIKI FUKUSHIMA, TOMOKAZU ISHIKAWA, Toyo University

ある。最後にネットワーク全体の損失は次のように与えられる。

$$L = L_Y + \lambda_1 L_O + \lambda_2 L_S \quad (4)$$

ここで、Zhao らはλ_1とλ_2のハイパーパラメータを探索せず、$\lambda_1 = \lambda_2 = 1$と設定しているため、本実験では$\lambda_1$と$\lambda_2$の値をそれぞれ $0.01, 0.1, 1, 10, 100$ の組み合わせで 25 通り実験を行うこととする。

4. 結果と考察

損失関数内のパラメータ探索結果で、最も性能が高かった場合と未調整の場合の比較結果を表 1 に示す。表 1 より、パラメータ調整を行うことで未調整の場合と比べ、約 5%精度が向上していることがわかる。

次に学習済みのモデルで VOT2018 データセット内にある Basketball の動画像内にいるバスケットボール選手の(図 1 中に緑色の矩形内の選手)について追跡を行った。

図 2a、図 2b より、$\lambda_1 = 1, \lambda_2 = 1$の場合に比べて、$\lambda_1 = 0.1, \lambda_2 = 10$の場合の方が追跡している物体にヒートマップが薄く広がっていることがわかる。このため、追跡を行う際に物体間のずれが生じにくくなることで、故障数が低減しやすくなると考えられる。

また、図 3a、図 3b より、$\lambda_1 = 1, \lambda_2 = 1$の場合と比べて、$\lambda_1 = 0.1, \lambda_2 = 10$の場合の方がバウンディングボックスのサイズが縮小しているが、図 1 より、今回追跡を行った物体はバスケットボール選手の頭頂部から膝周辺に設定しているため、$\lambda_1 = 0.1, \lambda_2 = 10$に設定した場合がより優れた結果を表示しているといえる。

次に Transformer の層を重ねた状態でパラメータ探索を行い、最も性能が高かった場合と未調整の場合の比較結果を表 2 に示す。

表 2 より、Transformer の層を重ねた場合であっても、パラメータ調整を行うことで未調整の場合と比べ、約 3%精度が向上していることがわかる。

表 1: Layer 数 1 で損失関数の値を探索した結果

Layer 1	Accuracy ↑	Robustness ↓	EAO ↑
$\lambda_1: 0.1, \lambda_2: 10$	**0.506**	**0.000**	**0.526**
$\lambda_1: 1, \lambda_2: 1$	0.457	0.308	0.429

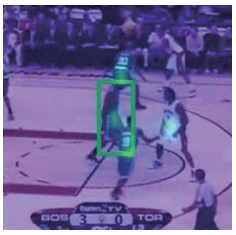
表 2: Layer 数 3 で損失関数の値を探索した結果

Layer 3	Accuracy ↑	Robustness ↓	EAO ↑
$\lambda_1: 0.1, \lambda_2: 10$	**0.511**	**0.000**	**0.520**
$\lambda_1: 1, \lambda_2: 1$	0.488	0.923	0.231

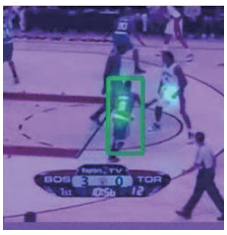
5. まとめと今後の課題

本研究では損失関数内のパラメータを調整することで既存研究のように未調整の場合と比べて精度が約 5%向上するという結果を得られた。損失関数内のパラメータを調整した際の結果は、Transformer の層を重ねた場合であっても同じように精度が向上する結果が得られるということがわかった。

図 1: 結果画像（追跡を行う選手を緑色のバウンディングボックスで囲んでいる）



(a) $\lambda_1 = 1, \lambda_2 = 1$



(b) $\lambda_1 = 0.1, \lambda_2 = 10$

図 2: 学習済みモデルを利用した際のヒートマップ付きサーチ画像

(a) $\lambda_1 = 1, \lambda_2 = 1$

(b) $\lambda_1 = 0.1, \lambda_2 = 10$

図 3: 学習済みモデルを利用した際の追跡結果画像

今後の課題は損失関数内の定数を変えた状態でバッチサイズを変えるとどのような変化があるのか確認することである。また、現在 Transformer のネットワーク内にある注意層は多くの論文で議論されている重要な構造である。そのため物体追跡のモデルにおいても、注意層を修正することで物体追跡の精度が向上することが見込まれるため、モデルを改良した実験を予定している。

参考文献

[1] Goutam Bhat, Martin Danelljan, Luc Van Gool, and Radu Timofte. Learning discriminative model prediction for tracking. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6182–6191, 2019.

[2] Moju Zhao, Kei Okada, and Masayuki Inaba. TrTr: Visual Tracking with Transformer. arXiv preprint arXiv:2105.03817, 2021.

[3] Hei Law and Jia Deng. Cornernet: Detecting objects as paired keypoints. In Proceedings of the European conference on computer vision (ECCV), pp. 734–750, 2018.