【ノート】 【令和2年度 先端技術等調査研究事業】

深層学習を用いた物体認識技術の高度化

高野 寛己，小泉 協*，小野 仁
機械電子情報技術部
（*現 新産業振興課）

産業界を中心に深層学習（Deep Learning）をはじめとしたAI技術の活用が著しい。深層学習を用いた画像処理の代表的技術として，画像分類，物体検出，領域分割（セグメンテーション）などが挙げられる。こうした深層学習技術は様々なタスクに対して，大変有用である場合が多いものの，学習時において大量の正解データが必要となること，評価時において予測結果の根拠が不明瞭であることなどが課題とされている。

当センターでは，こうした深層学習技術の抱える課題に対して有効な手法・手段について調査を行い，「正解データ作成時の工夫」や「予測の判断根拠箇所を可視化する技術」について開発・導入を試みた。

キーワード：AI，深層学習，Deep Learning，画像処理，アノテーション，Grad-CAM

1 緒言

深層学習は機械学習の一手法であるニューラルネットワークについて，多数の層を積み重ねたモデルを構築し，十分な学習データで学習を行うことで，複雑な表現や識別能力を持たせる技術である。深層学習は画像処理をはじめ，音声処理，言語処理と様々な分野で広く活用されている[1)]。特に画像処理分野の研究・開発において，深層学習は必要不可欠な要素技術となり，高い識別性能とその汎用性により，ますます発展を遂げている。深層学習を用いた画像処理の具体的な技術として，画像分類，物体検出，領域分割（セグメンテーション）がある（図1）。それぞれ画像全体に写っている物体を予測・識別する，画像上のどこに物体が写っているか示す，画像全体をピクセル単位で物体（物質）領域ごとにそれぞれ分割する手法である。

（a）画像分類 （b）物体検出[2)]

（c）領域分割（セグメンテーション）

図1 深層学習を用いた画像処理例

こうした深層学習を用いた画像処理は多様なタスクに対して有効であるため，様々な場面での活用が検討されているが，実際の現場で活用する際には，いくつか課題も挙げられる。本稿ではこれら課題について述べ，それらを克服するための工夫や技術について紹介する。

2 正解データ作成時の工夫

深層学習を用いた画像識別のタスクには，多数枚の正解データが必要となる。特にYOLO[2)]やSSD[3)]といった深層学習を用いた物体検出においては，学習データ作成時の際，画像上の物体の位置を指定するラベリング作業（アノテーション）が1枚ごとに必要となり，多数の画像データについて繰り返し同じような作業をすることは非常に煩雑である。

本章では，そうしたラベリング作業の煩雑性を低減させる工夫について紹介する。

2.1 オートアノテーション

動画データについてアノテーションする場合，動画を各フレームに分割し，1枚ずつラベリングする方法が考えられる。この際，検出対象が人間や動物，自動車といったある程度動きが予測できるものであれば，フレーム前後では大きな移動はほとんどないことが多い。そこで一枚目のフレームのみ人間の手作業または物体検出アルゴリズムでアノテーションを行い，それをもとに次フレームについては適切な画像処理技術を組み合わせることで，機械的に正解データを作成できると考えられる。

2.2 手法

本節では，野生動物であるシカ動画のアノテーション作業について述べる。あらかじめ20～30枚程度手作業でアノテーションした画像を用意し，シカをある程度検出できる学習済みモデルを作成する。学習枚数が少ないため，検出漏れや誤検出などが多いモデルとなる。これを「事前学習」とする。

次に事前学習で生成した学習済みモデルを活用し，学習データ枚数を大幅に増やすことを目指す。図2は30秒程度のシカ動画をフレーム毎に画像に分割した1枚目のフレームである。図2のシカを囲んだ矩形領域を追従領域とし，その周囲について縦横416pixels四方を探索領域とする。（参考：YOLO v3[2]の入力画像サイズは416pixels。）この探索領域について，次フレームの同領域をYOLOに入力し，シカが検出された場合は，その位置情報を学習データとして保存する。なお前述のとおり，事前学習では，検出漏れや検出箇所のずれが大きいため，タスクにもよるが正しく検出できないものも多い。

探索領域においてシカが検出されなかった場合は，前フレームの探索領域内の追従領域周囲で時間差分をとり，変化が大きかった方向へ追従領域を移動させ，当フレームの学習データとする。こうして新しい学習データを機械的に作成することが可能である。ここでは，追従領域周囲の差分画像を2値化し，変化した画素数が多い方向へ追従領域を移動させている。なお画素数にあまり差がない場合は移動せず，追従領域は前フレームと同じ座標位置と設定する。

こうして得られた学習データについて，最後に人間の目視により，明らかに物体位置を示していない学習データの除去を行い，それらを用いて「本学習」を行う。

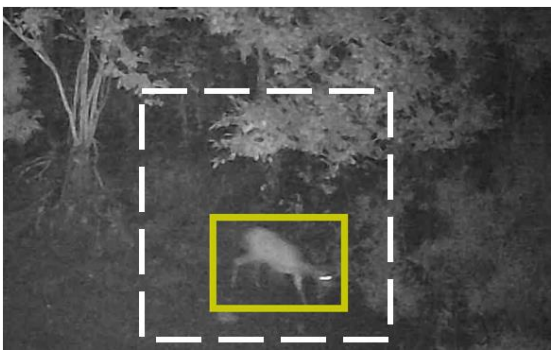

図2 動画フレームの一例

（白点線枠：探索領域，黄実線枠：追従領域）

3 予測の判断根拠箇所を可視化する技術

深層学習の中身はブラックボックスと呼ばれることが多く，出力や予測結果の判断根拠を示すことは一般的には難しい。この課題を解決すべく様々な技術が開発検討されている[4]。今回の調査では参考文献4において判断根拠の可視化に有効と見なされた手法の一つである，Grad-CAM[5]の調査を行い，導入を試みた。

3.1 Grad-CAMの原理

(ア) CAM[6]について

図3のように畳み込みニューラルネットワーク（CNN）の最終出力層は，各クラス（カテゴリ）の確率値となる。一つのクラスcに着目した場合，cは前段の重みそれぞれw_1，w_2,$\cdots w_n$と，最終畳み込み層の各特徴マップの平均値との線形結合で表される。最終ニューロン（予測クラス）は確率値であり，例えば重みの数値が大きいほど確率値も大きくなる。よって前段ニューロンの重みの大きさが，ある予測クラスの判別に重要であると分かる。

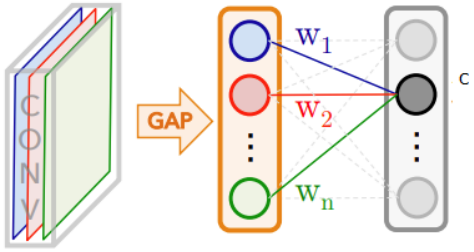

図3 CNNの全結合層[6]

最終畳み込み層のk番目の特徴マップ$f_k(x,y)$について，特徴マップ上の座標(x,y)と特徴マップをプーリングした出力をF_kとすると，式(1)となる。

$$F_k = \sum_{x,y} f_k(x,y) \tag{1}$$

式(1)を用いて，最終ニューロンのクラスcの出力y_cは式(2)で表される。

$$y_c = \sum_k w_k \sum_{x,y} f_k(x,y) = \sum_{x,y}\sum_k w_k f_k(x,y) \tag{2}$$

式(2)の右辺$\sum_k w_k f_k(x,y)$はクラスcにおけるCAMを表しており，これを新たに$M_c(x,y)$と置くと，

$$y_c = \sum_{x,y} M_c(x,y) \tag{3}$$

となる。すなわち$M_c(x,y)$はクラスcのCAM画像であるから，CAM上の座標がクラスcの判断根拠箇所を指すことが数式的に示された。

(イ) Grad-CAM

Grad-CAMは前項の式(2)の重みw_kについて，誤差逆伝播適用時の勾配α_k^cを活用したものである。勾配α_k^cは最終ニューロン(クラスc)の出力値をy_c，k番目の特徴マップの出力$f_k(x,y)$を用いて，式(4)で表される。

$$\alpha_k^c = \frac{1}{Z}\sum_x\sum_y \frac{\partial y_c}{\partial f_k(x,y)} \tag{4}$$

ここでZは規格化定数$Z = \sum_{x,y} 1$である。

クラスcにおけるGrad-CAMをL_cとおくと，式(5)となる。

$$L_c = ReLU\left(\sum_k \alpha_k^c f_k(x,y)\right) \tag{5}$$

ここでReLU(Rectified Linear Unit)関数は式(6)で表される活性化関数であり，ニューラルネットワークの各レイヤーの出力時に非線形変換をするうえで必要となる。

$$ReLU(x) = \begin{cases} x & (x \geq 0) \\ 0 & (x < 0) \end{cases} \tag{6}$$

$\frac{\partial y_c}{\partial f_k(x,y)}$は出力値$y_c$を特徴マップ$f_k(x,y)$で微分したものである。ある座標$(x,y)$において，この値が大きくなった場合，特徴マップ上のその位置が予測クラスcに大きく影響したことが分かる。

3.2 実験結果

実装は機械学習のフレームワークの一つであるTensorflow2.4.0を導入し，python3.7を用いて開発した。統合開発環境としてSpyder(Anaconda3)を用いた。

ハードウェアのスペックは以下の通り。

- CPU: Intel Core i7-9700K CPU
- RAMメモリ:48GB
- GPU: NVIDIA GeForce GTX1080Ti

今回使用した画像データセットは自然画像の花のデータセットであり，ヒナギグ，タンポポ，バラ，ヒマワリ，チューリップ(英名はそれぞれdaisy，dandelion，rose，sunflower，tulip)の5クラス分類を行った。

訓練画像は各クラスで約500～700枚程度，評価画像はそれぞれ50枚を用いた。学習に使用したニューラルネットワークはMobileNet V2[7]であり，学習手法として，あらかじめimagenet(大規模自然画像データセットの一つ)で学習したパラメータを転用し，今回の実験に応じて変更した最終層を20epoch程度学習する，転移学習を行った。その結果性能として，正解率85%程度の識別能力となった。こうして得られた学習済モデルを用いて，画像の評価を行いGrad-CAMの出力を得た。

図4のGrad-CAM画像の出力はOpenCV4.4.0のCOLORMAP_JETを使用した。赤色に近いほど注視領域であり，図4の結果よりニューラルネットワークが，予測の際に花びら付近を注視していることが確認された。なお各図上の正解及び予測については，それぞれ画像の正解ラベルと予測ラベルを表している。

図4 ニューラルネットワークの注視領域
（左） 元画像 （右） Grad-CAM画像

4 結言

深層学習を用いた画像処理技術の現状の課題について調査を行い，解決策となる手法や技術について開発・導入を試みた。本調査で導入したオートアノテーションの手法を用いることで，データ作成段階の冗長作業の軽減が期待される。また評価予測時の際，Grad-CAM等の画像上の注視領域を可視化する技術を用いることで，予測結果に対する説明可能性が向上すると考えられる。

参考文献

1) 独立行政法人情報処理推進機構:AI白書2020，角川アスキー総合研究所，2020
2) J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in 2016 IEEE Conference on CVPR.
3) W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “SSD: Single shot multibox detector,” in Computer Vision - ECCV 2016, pp. 21-37, Springer International Publishing, 2016.
4) J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim, “Sanity Checks for Saliency Maps Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization” in 2018

NeurIPS.

5) R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D.Parikh, and D. Batra, “Grad-cam: Why did you say that?”, arXiv preprint arXiv:1611.07450, 2016.

6) B. Zhou, A. Khosla, A. Lapedriza, A. Olivia, A. Torralba, “Learning Deep Features for Discriminative Localization”, in CVPR 2016.

7) M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.Chen, ”MovileNetV2: Inverted Residuals and Linear Bottlenecks”, arXiv:1801.04381v4, 2019.