令和 3 年度

修士学位論文

少データ・少ラベル学習のためのデータ拡張法と農業画像認識への応用

A Data Augmentation Method for Few-Shot and Few-Annotation Learning and Its Application to Agricultural Object Detection

1245134　　野田 峻平

指導教員　　吉田 真一

2022 年 2 月 28 日

高知工科大学大学院 工学研究科 基盤工学専攻

情報学コース

要 旨

少データ・少ラベル学習のためのデータ拡張法と農業画像認識への応用

野田 峻平

近年、農業人口の減少や不安定な供給を考慮して，農作物の自動収穫や自動管理などの効率的な農業が期待されており，自動化を実現するためには農作物の位置情報の取得が不可欠である．位置情報の取得には、ニューラルネットワークによる物体検出が用いられているが，ニューラルネットワークの学習には，高品質かつ大規模なデータセットが必要であり，データセット作成のためのコストが大きい．そこで，学習データの画像数を減らすことを少データ (Few-shot) 設定，アノテーション数を減らすことを少ラベル (Few-annotation) 設定としてデータセットに導入し，このデータセットを用いて物体検出モデルを構築する方法として，Few-shot 用の物体検出モデルとデータ拡張手法を用いる手法を提案する．提案手法では少数の物体領域画像と多数の背景領域画像から新たにデータセットを作成し，学習させることで Few-shot モデルを構築する．Few-shot 設定のみ導入したデータセットで学習させた比較用モデル Faster R-CNN と Few-shot モデルと Few-shot 設定と Few-annotation 設定を導入したデータセットで学習させた提案手法を比較する．結果として，提案手法の AP50 は比較用 Few-shot モデルより 0.4 から 4.8 ポイント低く，比較用 Faster R-CNN モデルより 5.8 から 18.0 ポイント高い．また，提案手法では他の手法よりアノテーションの数を 69%から 94%減少する．そして，学習に必要な時間は Few-shot モデルを用いたことにより，Faster R-CNN よりも 90%削減できる．

キーワード　物体検出, 少データ・少ラベル学習, データ拡張

Abstract

A Data Augmentation Method for Few-Shot and Few-Annotation Learning and Its Application to Agricultural Object Detection

NODA Shumpei

Automation of agricultural management has become important by an aging population and decreasing agricultural population. A deep neural network can be applied to detect vegetables and fruits on a farm. Detecting objects in an image taken from a camera leads to counting the objects. However, the training cost for the deep neural network is generally high, and it requires a high-quality large dataset and time consumption. Also, there are various kinds of vegetables, and it is not feasible to train and tune neural networks for all of them because creating datasets for all vegetables requires a high cost. Then, we propose the method which employs a few-shot model and data augmentation for few-shot and few-annotated datasets. Finally, we create the novel dataset for detecting eggplants as a dataset for our experiment and evaluation. As a result, the proposed method with the few-annotation and few-shot settings obtain from 5.8 to 18.0 points improvement of accuracy than that of the Faster R-CNN with the few-shot settings on AP50. The proposed AP50 is lower 0.4 to 4.8 points than that of the few-shot model. with the few-shot setting. The proposed reduces the number of annotations by 69% to 94%. Then, the proposed takes 90% fewer times than others compared.

key words Object detection, Few-shot and Few-annotation Learning, Data aug-

mentation

目次

目次

図目次

表目次

第 1 章

序論

日本の農業現場では，農業従事者の不足と高齢化が問題である．農業従事者の数は年々減少しており，平成 27 年に 175.7 万人であった農業従事者の数は令和 3 年には 130.2 万人にまで減少している [1]．また, 農業従事者の高齢化が進み，平均年齢が令和 2 年には 67.8 歳となり，平成 27 年から令和 2 年の間でもっとも高い値となった [1]．これらの問題があるため，農業に IT を導入し，解決することが期待されている．

現在行われている IT 技術の一つである IoT を用いた農業では，温度センサーや湿度センサーを用いた自動的な環境の調節やドローンを用いた広範囲への農薬の散布などが行われている．しかし，これだけでは農業全体の自動化を行うことはできない．自動化には農作物に対する作業が重要であり，それは農作物の自動的な収穫や管理がある．それらを実現するためには農作物の位置情報の取得が重要となる．

農作物の位置情報は物体検出手法を用いることで取得することができる．そして，物体検出手法にはさまざまな方法があるが，その中でも近年精度が大きく向上し，広く利用されているものにディープラーニング，AI を用いた方法がある．AI の学習ではタスクの入力値と入力に対して期待する出力値を，教師データとしてデータセットとして複数作成し，学習用データセットを構築することで，アルゴリズムによって自動的に学習する．さまざまなデータやタスク (処理する問題) に対して AI モデルとデータセットを用意するだけで簡単に使用することができるが，予測精度を求める場合は多くの質の良いデータが必要となる．そのため，データセットの作成には多くの時間や労働力が必要となる．データセット作成におけるコスト削減のために Fine-tuning や Few-shot Learning といった手法が提案され，少ないデータで AI モデルを様々な環境に適用できる．これらの手法ではあらかじめ公開されてい

る既存の大きなデータセットを用いて学習を行い，その学習済みの AI モデルを目的のデータセットでさらに学習させることで，別タスクでの学習で得られた精度を保ったまま，より小さな目的のデータセットへと適用することができる．しかし，データセットを小さくした場合においても AI モデルを適用する農業環境それぞれに対してデータセットを作成する必要があり，アノテーションコストが大きい．また，物体検出タスクは 1 つの検出場面から複数の物体を一度に検出するため，作成するデータセットでのアノテーション (教師情報付与) 作業が多く，データセット作成のコストが他の画像認識タスクと比べて大きい．そのため，農業分野での AI の活用には，教師データの画像の数を減らし，アノテーションの数も減らして学習できる手法が必要となる．

本研究では，物体検出用の学習データセットに含まれる画像の数を減らすことを少データ (Few-shot) 設定とし，アノテーションの数を減らすことを少ラベル (Few-annotation) 設定とする．農業ハウス内から物体としてナスを検出するためのデータセットを農業用データセットとして作成し，このデータセットを AI モデルで学習させることでナスを検出するための AI モデルを構築する．このデータセットに Few-shot 設定と Few-annotation 設定を導入し，農業ハウス内の画像とナスの教師ラベルを減らすことで，擬似的に少ないデータで構成されるデータセットを作成する．そして，Few-shot 設定のために提案された AI モデルと Few-annotation 設定のために提案するデータ拡張手法を用いて 2 つの設定が導入されたデータセットから AI モデルを構築する．Few-shot 設定のみが導入されたデータセット，Few-shot 設定が考慮されていない AI モデルなどを用いて提案手法の効果を指標 Average Precision を基準に比較する．

第 2 章

関連技術

本研究では畳み込みニューラルネットワークベースの物体検出手法 (ニューラル物体検出) を用いて物体の検出を行う．そして，ニューラル物体検出用モデルは入力画像の特徴抽出から物体位置の推定までの全てをニューラルネットワークで行うモデルを用いる．画像の数が少ない状態，Few-shot 設定がある状態でモデルを学習する手法は Few-shot Learning として既存の研究が存在する．また，ニューラルネットワークでのデータ拡張手法はアノテーションの数を人工的に増やすことを行うので，Few-annotation 設定がある状態でのモデルの学習に応用することができる．これらのことから，畳み込みニューラルネットワークとそれを用いた特徴抽出器，物体検出モデルを説明し，最後に Few-shot 設定と Few-annotation 設定に関連する Few-shot Learning とデータ拡張について説明する．

2.1 畳み込みニューラルネットワーク (Convolutional Neural Network)

畳み込みニューラルネットワーク (Convolutional Neural Network, CNN) は主に画像処理タスクに用いられるが，近年では自然言語処理タスクにも用いられる．CNN モデルは畳み込み層とプーリング層を組み合わせて構成し，畳み込み層の畳み込みカーネルが学習することで予測を可能としている．畳み込み層では畳み込みカーネルを用いて画像内の複数画素を 1 つにまとめ，入力画像の物体の位置のズレや変化に対応する．また，プーリング層では最大値プーリングを用いることで複数画素の中から重要な値となる画素を抽出する．物体検出モデルを構築する際，CNN は画像からの特徴量抽出などに用いられる．本研究において

は CNN やニューラルネットワークを組み合わせて 1 つのモデルを構築することで，物体検出用のニューラルネットワークモデルを作成する．

2.2　特徴抽出器

物体の分類，検出タスクや文章の判定タスクなどでは RGB 画像や自然言語を入力として与える．このとき，画像や自然言語に対して前処理を行わずに入力し，タスクに適用する場合，全ての情報が等価に扱われるため，必要な情報とそうでない情報の判別が付きづらい状態である．そのため，機械学習モデルにこれらの情報入力をした場合，学習が困難になる可能性がある．そこで，入力された情報の重要性に応じて表現を変化させることで対応する．機械学習では入力データから特徴量の抽出を行うが，人間がアルゴリズムを考え，それに応じた特徴量の抽出を行う特徴工学的手法と特徴抽出方法も機械学習によって獲得する表現学習がある．近年，ニューラルネットワークを用いた画像認識の分野では特徴量の抽出からその特徴量を用いた予測までを 1 つのニューラルネットワークモデルで行う．このディープニューラルネットワークを用いて特徴量抽出を行うためのモデルのことを特徴抽出器とする．本研究ではニューラルネットワークを用いた物体検出手法を用いるとき，特徴抽出器として Residual Networks (ResNet) [2] を用いる．また，物体検出において小さい物体の画像領域から得られる特徴量と大きい物体の画像領域から得られる特徴量の属性の違いが発生することに対処するため，Feature Pyramid Network (FPN) を用いる．

2.2.1　Residual Networks (ResNet)

一般的に，ニューラルネットワークは層を深くすることで学習をするためのパラメータ数が増加し，表現の幅が大きくなる．そのため，ディープラーニングなどでは予測モデルの層を増やすことでタスクに対しての精度を増加させている．しかし，層が深くなることによって学習時に行われる誤差逆伝搬が浅い層でできなくなるため，学習が困難となる．この層を増やしたときの問題を解決し，152 層もの学習レイヤーを持つディープニューラルネット

ワークモデルに ResNet がある．ResNet は shortcut connection と呼ばれる複数の層のスキップとスキップした層全体の最終出力との足し合わせを行うことによって，スキップした層ではスキップする前の入力値と出力値に対する差を学習するようになる．この残差を計算する過程で行われる逆伝搬はアンサンブル学習や勾配ブースティング法に似ているため，逆伝搬時の微分計算によって勾配が消失しにくい計算となっている [3].

2.2.2　Feature Pyramid Networks (FPN)

農業ドメインにおける物体検出では物体のスケールの違いがある．このような場合に Feature Pyramid を用いてスケールの異なる物体に対して有効な特徴量を作成する．Feature Pyramid の作成は異なるアスペクト比に対応する特徴抽出器をアスペクト比の数だけ用意する方法や多層の特徴抽出器の中間層が出力する特徴量をそれぞれのアスペクト比に対応させる方法がある．しかし，特徴抽出器をアスペクト比の数だけ用意する場合，モデルの学習コストが高い．また，中間層の特徴量を用いる場合，出力した層によって表現力の違いが発生する．そこで FPN では，1 つの多層ニューラルネットワークの中間層の特徴量を表現力が同じになるように，低次元の表現力を持つ特徴量と高次元の表現力を持つ特徴量を合成することによって同じ表現力を持ち，対応スケールが異なる特徴量を生成する [4].

2.3　ニューラル物体検出

CNN ベースの物体検出では，入力された画像に対して指定したクラスの物体の検出を行い，出力として物体の位置とクラスの信頼スコアを出力する．物体検出モデルの学習には画像とアノテーションのペアを用いる．アノテーションは物体の位置を示すバウンディングボックスとクラスラベルを含む．CNN ベースの物体検出手法は大きく分けて One-stage モデルと Two-stage モデルの 2 種類がある．One-stage モデルには You Only Look Once (YOLO) [5] や Single Shot Multibox Detector (SSD) [6] といったモデルが存在し，これらのモデルでは画像から抽出した特徴量から直接的に物体の位置とクラスを予測する．

Two-stage モデルには Region-based Convolutional Neural Network (R-CNN) [7], Fast R-CNN[8], Faster R-CNN[9] といったモデルがあり，これらのモデルでは 1 段階目に物体らしい領域の予測を行い，2 段階目で最終予測を行う．特に Faster R-CNN では，1 段階目では画像から特徴量を抽出し，その特徴量から物体らしい領域を予測する．2 段階目では 1 段階目で予測した物体らしい領域に対応する画像の部分特徴量を抽出し，その特徴量から 1 段階目に予測した位置からの修正値とクラスラベルを予測することで最終予測を行う．一般的に Two-stage モデルは One-stage モデルの予測精度より高い予測精度を安定的に持ち，One-stage モデルは Two-stage モデルより予測の処理が軽量である．近年では One-stage モデルの予測精度も向上してきている．また，一般的に物体検出モデルの精度の指標としては Average Precision を用いる．

2.3.1　Faster Region Convolutional Neural Networks (Faster R-CNN)

Faster R-CNN は R-CNN と Fast R-CNN を入力から出力まで全てニューラルネットによって行うモデルに改良し，処理の高速化を行ったモデルである．R-CNN から Fast R-CNN の改良では物体らしい領域の特徴量の抽出回数を減らすことによって改良を行った．Fast R-CNN から Faster R-CNN への改良では物体らしい領域の抽出方法を Selective Search でなく，ニューラルネットワークによる生成をすることによって改良を行った．Faster R-CNN は特徴抽出器，Region Proposal Network (RPN)，Region of Interest Pool (RoIPool) または Region of Interest Align (RoIAlign) [10] から構成される．

2.3.2　Region Proposal Network(RPN)

RPN は入力画像の特徴量から物体らしい領域を予測する．予測にはアンカーと呼ばれる異なるスケールとアスペクト比を持つ矩形を複数作成し，画像に網羅的に配置をする．RPN は各アンカーに対して物体らしい領域との差を予測することで物体らしい領域を予測する．

そのため，RPN でのニューラルネットワークによる予測ではアンカーの数だけの座標の差を回帰タスクとして求め，物体らしいかどうかを 2 値分類タスクとして求める．このとき，複数のアンカーが 1 つの物体を示すようにアンカーとアンカーとの重なりが存在する場合がある．そこで，同じ物体に対する複数の予測を制御するために Non-maximum supression (NMS) を用いて不要な予測を削除する．その後，物体らしいと判定した領域を候補領域として 2 段階目の処理へ渡す．

2.3.3　Non-Maximum Suppression(NMS)

NMS は物体位置の予測領域の重なりが発生した時に，その予測領域が同一の物体に対する予測を行っているのか別の物体に対して予測を行っているのかを判定する．物体のクラスごとに分けられた予測領域をスコアの降順で並べ替え，スコアの高いものから選択し，選択した領域とある一定割合の大きさで重なっている領域を同じ物体に対する予測とみなし，削除することによって同じ物体に対する予測を抑制する．領域の重なり度合いを判定するときには Intersection over union(IoU) を用いる．

Intersection over Union (IoU)

IoU は 2 つの領域の重なりの度合いを示す．物体検出では領域と領域の重なりの度合いを測るために用いる．2 つの領域 B_0 と B_1 があり，重なっている領域を B_2 とした時，IoU は

$$\mathrm{IoU} = \frac{B_2}{B_0 + B_1 + B_2} \tag{2.1}$$

となる．

2.3.4　Region of Interest Pool (RoIPool)

RoIPool は物体らしい候補領域に対応する部分特徴量を固定サイズで抽出するための手法である．候補領域の座標を整数値に丸め，特徴量の大きさに合わせて対応する座標へと当てはめる．当てはめた領域を任意の大きさのグリッド状に分割することで，複数のセルを作成

表 2.1 物体検出の予測結果の例．Rank は予測スコアを基準にした順序を示す．Correct は予測結果が正しいかどうかを示す．

Rank	Correct	Precision	Recall
1	✓	$1/1 = 1.00$	$1/3 = 0.33$
2		$1/2 = 0.50$	$1/3 = 0.33$
3		$1/3 = 0.33$	$1/3 = 0.33$
4	✓	$2/4 = 0.50$	$2/3 = 0.67$
5	✓	$3/5 = 0.60$	$3/3 = 1.00$

する．各セル内の値の平均値や最大値をセルの代表値とし，全てのセルの持つ値を固定サイズの特徴量とする．

2.3.5 Region of Interest Align (RoIAlign)

RoIAlign は RoIPool で候補領域の座標を丸めたときに発生する誤差の悪影響を改善した手法である．RoIAlign では，候補領域を特徴量の対応する座標へ当てはめるときに，座標を丸めずに当てはめる．そして，当てはめた領域を任意の大きさのグリッド状に分割することで，複数のセルを作成する．各セルの中の角に 4 点を取り，各点の値を 4 近傍画素のバイリニア補完法によって求める．求めた各点の値の平均値や最大値をセルの代表値とし，全てのセルの持つ値を固定サイズの特徴量とする．本研究で用いる物体検出モデルでは RoIAlign を導入したモデルを構築する．

2.3.6 Average Precision(AP)

AP は物体検出において，検出間違いと検出漏れがないように物体を検出することで高い値となる指標である．AP の計算方法を例を用いて説明する．表 2.1 は検出対象物体が画像内に 3 つ含まれ，正しい予測と間違った予測の 5 つを行なった場合の Precision と Recall の値を示す． 図 2.1 は表 2.1 の Precision と Recall の曲線を示す． AP の計算では予測結果を予測スコアの降順に並べ，各 Rank にまでの Precision と Recall を順番に計算する．Rank が 2 の場合，Rank2 までで正しく予測できた物体の数は 1 つであり，検出対象物体 3

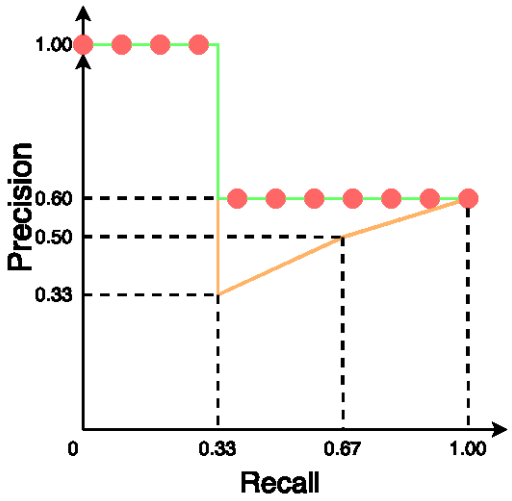

図 2.1 Precision と Recall の曲線例．オレンジ線は補完前の曲線，緑線は補完後の曲線，赤点は平均値を求める際に用いる各値を示す．

つの内 1 つを検出できているため，Precision は 0.50 となり，Recall は 0.33 となる．この計算を Recall が 1 になるまで，または予測結果に対して全て計算するまで行う．Precision と Recall から曲線 (図 2.1 オレンジ線) を作成し，その曲線を補間した曲線 (2.1 緑線) を作成する．最後に図 2.1 に示した補完した曲線の 11 点 (赤点) の平均値を AP とする．また，AP50 は予測領域と正解領域の IoU が 50%以上の時を正解と判定した場合の AP であり，AP75 は IoU75%以上の時を正解と判定した場合の mAP である．そして，mAP は予測領域と正解領域の正解と判定する IoU を 5 ポイント幅で 50%から 95%まで変化させた時の結果の平均 AP を表す．

2.4 Few-shot Learning

Few-shot Learning について説明する．Few-shot Learning はニューラルネットワークモデルを各クラス少ないデータで学習するための手法である．この手法は新しいクラスを追加するときに，より少ないデータセットで適用可能にすることを目的としている．物体検出においては，モデルは各クラスにおいて少数の画像と各画像に含まれるアノテーションから学習する．Few-shot 設定において，いくつかの手法では各クラスの物体の例画像の集合や検

出対象を表す例画像 1 枚をモデルに入力として与え，例画像に対応する物体を検出するタスクを行うことで，一般物体検出データセットに含まれにくい珍しい物体や含まれていない未出現の物体を検出する．Kang らや Perez らによって提案された手法 [11] や [12] においては，物体の例画像の集合から得られた各クラスの重みで検出場面画像の特徴量に重み付けをすることで予測結果を調整し，検出対象の全クラスに対する予測を行う．Michaelis ら，Fan らや Hsieh らによって提案された手法 [13]，[14] や [15] においては，物体の例画像 1 枚から抽出した特徴量と検出場面画像から抽出した特徴量に注意機構を用いることで，双方向的に関係のある部分に対して重み付けを行い，例画像が示す物体 1 クラスに対する予測のみを行う．

2.5　データ拡張 (Data augmentation)

一般的にニューラルネットワークモデルを多くのデータで学習させることは汎化性能の向上に影響を与える．しかし，人的労働力によるデータセットの作成には大きなコストを要求するため，大きなデータセットを作成し，学習させることは難しい．データ拡張では既存のデータから人工的かつ自動的に新たなデータを作成する．そして，既存データと拡張用データを用いてニューラルネットワークモデルを学習させることで，モデルの正則化に影響を与える．画像のクラス分類タスクにおいて，MixUp[16] は二枚の画像を合成し，その合成割合に応じたクラススコアの予測を行うことで，二つのクラスの中間のクラスなどを予測できるようにしている．また，CutOut[17] や Random erasing[18] では，分類を行う画像の一部部分にマスクをかけた状態で予測を行うことで，全体が見えていない物体に対しても予測ができるようにしている．物体検出タスクにおいて，Copy-paste augmentation[19] や Patch-level augmentation[20] はある学習用画像に含まれる物体を別の学習用画像に対して貼り付けることで学習データ全体の物体の数や検出場面画像のコンテキストを増やすことができる．本研究では汎化性能の向上を目的としたデータ拡張を行うと同時に，Few-annotation 設定を導入したデータセットのアノテーション数の増加とノイズとなる

2.5　データ拡張 (Data augmentation)

データの削除のためにデータ拡張を用いる.

第 3 章

提案手法

Few-shot 設定と Few-annotation 設定を導入したデータセットで物体を検出するための手法を提案する．Few-shot 設定に対応するために Few-shot 設定を考慮したモデルを構築し，Few-annotation 設定に対応するために Few-annotation 設定が導入されたデータセットから新しいデータセットを作成するデータ拡張手法を提案する．パッチワーク拡張では提案データ拡張手法で作成するパッチワーク画像の具体的な作成方法を説明し，パッチワーク画像がどのようなものであるかを示す．Few-shot 設定を考慮したモデル (Few-shot モデル) ではモデル構造を示し，モデルの内部で行われる具体的な処理と目的について示す．

3.1 パッチワーク拡張

Few-annotation 設定をデータセットに導入してアノテーションの数を減らしたとき，画像内に含まれる物体領域にはアノテーションがあるものとないものの 2 つに分かれる．このデータセットでモデルを学習すると，アノテーションがない物体領域が背景領域として扱われるため，ナスの物体情報をモデルが背景として学習し，モデルがこのデータセットにアンダーフィットする可能性が高い．また，アノテーションのある物体領域のナスのみをナスの物体情報としてモデルが学習するため，モデルが過学習する可能性もある．そこで，Few-annotation 設定のあるデータセットからアノテーションのない物体領域を含まない新しいデータセットを作成する方法であるパッチワーク拡張を提案する．

学習データセットの N_{I} 枚の $H_{\mathrm{I}} \times W_{\mathrm{I}}$ の画像を $I \in \mathbb{R}^{N_{\mathrm{I}} \times 3 \times H_{\mathrm{I}} \times W_{\mathrm{I}}}$，$N_{\mathrm{B}}$ 個の物体領域を示すバウンディングボックスを $B \in \mathbb{R}^{N_{\mathrm{B}} \times 4}$ とする．バウンディングボックス B の i 番目

の要素は $b_i = \{x_i, y_i, w_i, h_i\}$，$b_i \in B$ であり，x_i は物体領域の始点の x 座標，y_i は物体領域の始点の y 座標，w_i は物体領域の横幅，h_i は物体領域の縦幅を表す．

作成するデータセットに含まれる画像のことをパッチワーク画像と呼び，小さな複数の画像パッチを繋ぎ合わせて 1 枚を作成する．画像パッチを縦に h_{g} 枚，横に w_{g} 枚繋ぎ合わせたパッチワーク画像をグリッドサイズ $h_{\mathrm{g}} \times w_{\mathrm{g}}$ のパッチワーク画像 I_{pw} とする．また，パッチワークに対応するバウンディングボックスを B_{pw} とする．

$S_{\mathrm{pw}} \times S_{\mathrm{pw}}$ の N_{pw} 枚のパッチワーク画像 $I_{\mathrm{pw}} \in \mathbb{R}^{N_{\mathrm{pw}} \times 3 \times S_{\mathrm{pw}} \times S_{\mathrm{pw}}}$ を作成する場合の説明を行う．このときの画像パッチのサイズ $H_{\mathrm{p}} \times W_{\mathrm{p}}$ は

$$H_{\mathrm{p}} = S_{\mathrm{pw}} / H_{\mathrm{g}} \tag{3.1}$$

$$W_{\mathrm{p}} = S_{\mathrm{pw}} / W_{\mathrm{g}} \tag{3.2}$$

となる．グリッドサイズは N_{pw} 個あり，i 番目の h_{g_i} を縦のグリッドサイズ，w_{g_i} を横のグリッドサイズとしたとき，グリッドサイズは $S_g \in \mathbb{R}^{N_{\mathrm{pw}} \times 2}$，$s_i = \{h_{\mathrm{g}_i}, w_{\mathrm{g}_i}\}$ となる．パッチワーク画像の作成では，物体領域を表す画像パッチである Positive パッチ p_{pos} と背景領域を表す画像パッチである Negative パッチ p_{neg} を用意し，パッチ画像としてランダムに選択し，繋ぎ合わせる．

はじめに N_{neg} 枚の Negative パッチ P_{neg} を作成する方法を説明する．I から固定サイズ $S_p \times S_p$ のパッチ画像をランダムに $\tilde{N}_{\mathrm{neg}}$ 枚抽出し，Negative パッチ候補 $\tilde{P}_{\mathrm{neg}} \in \mathbb{R}^{\tilde{N}_{\mathrm{neg}} \times 3 \times S_{\mathrm{p}} \times S_{\mathrm{p}}}$ とする．Negative パッチ候補はランダムに抽出していることから，背景領域だけでなく，物体領域を含む可能性がある．そこで，k-means による分類で Negative パッチ候補を物体領域のパッチ画像と背景領域のパッチ画像の 2 つに分ける．この分類を行うために用いる $\tilde{N}_{\mathrm{pos}}$ 枚の Positive パッチ $\tilde{P}_{\mathrm{pos}} \in \mathbb{R}^{\tilde{N}_{\mathrm{pos}} \times 3 \times S_p \times S_p}$ を作成する．$\tilde{P}_{\mathrm{pos}}$ はバウンディングボックス B が表す物体領域を画像 I から抽出することで作成する．また，複数スケールの Positive パッチ作成するために，1 つの Positive パッチ $\tilde{p}_{\mathrm{pos}} \in \tilde{P}$ を各グリッドサイズに対応するパッチ画像のサイズにリサイズし，$\tilde{N}_{\mathrm{pos}} = N_{\mathrm{pw}} \times N_{\mathrm{B}}$ 枚の Positive パッチを作成する．作成した Positive パッチ $\tilde{P}_{\mathrm{pos}}$ と Negative 候補パッチ $\tilde{P}_{\mathrm{neg}}$ を k-means によって N_{k} のクラスタに分ける．最後に Positive パッチと同じクラスタに分類された

Negative パッチ $\hat{P}_{\mathrm{neg}}$ を Negative パッチ候補 $\tilde{P}_{\mathrm{neg}}$ から取り除くことで Negative パッチ $P_{\mathrm{neg}} = \tilde{P}_{\mathrm{neg}} \backslash \hat{P}_{\mathrm{neg}}$ とする．

次にパッチワーク画像を作成するための Positive パッチ p_{pos} を作成する方法を説明する．この Positive パッチはパッチワーク画像に使用するため，物体領域の周辺領域も含んだ画像とする．周辺領域の大きさを表すマージンを $m = \{w_{\mathrm{l}}, h_{\mathrm{l}}, w_{\mathrm{r}}, h_{\mathrm{r}}\}$ とし，w_{l} は物体領域の左側のマージン，h_{l} は物体領域の上側のマージン，w_{r} は物体領域の右側のマージン，h_{r} は物体領域の下側のマージンを表す．また，i 番目のバウンディングボックスに対応するマージンを $m_i = \{w_{\mathrm{l}i}, h_{\mathrm{l}i}, w_{\mathrm{r}i}, h_{\mathrm{r}i}\}$ としたとき，i 番目のバウンディングボックス b_i に適用したときの抽出範囲を表すバウンディングボックス $\tilde{b}_i = \{\tilde{x}_i, \tilde{y}_i, \tilde{w}_i, \tilde{h}_i\}$ は

$$\tilde{x}_i = x_i - w_{\mathrm{l}i} \tag{3.3}$$

$$\tilde{y}_i = y_i - y_{\mathrm{l}i} \tag{3.4}$$

$$\tilde{w}_i = w_i + x_{\mathrm{l}i} + x_{\mathrm{r}i} \tag{3.5}$$

$$\tilde{h}_i = h_i + y_{\mathrm{l}i} + y_{\mathrm{r}i} \tag{3.6}$$

となる．マージンはそれぞれの要素がとりうる最大値を設定し，その最大値までのランダムな値をとることで決定する．抽出範囲を表すバウンディングボックス $\tilde{b}$ に対応する領域画像を画像 I の中から抽出することで Positive パッチ p_{pos} とする．

最後にパッチワーク画像 $I_{\mathrm{pw}} \in \mathbb{R}^{3 \times S_{\mathrm{pw}} \times S_{\mathrm{pw}}}$ と Positive パッチに対応する $\tilde{N}_{\mathrm{B}}$ 個バウンディングボックス $B_{\mathrm{pw}} \in \mathbb{R}^{\tilde{N}_{\mathrm{B}} \times 4}$ を作成する方法を説明する．パッチワーク画像は Positive パッチと Negative パッチをランダムに選択し，繋ぎ合わせることで作成する．Positive パッチは選択した時点で生成し，Negative パッチは P_{neg} からランダムに選択する．Positive パッチ p_{pos} の作成を選択する確率を $\bar{p}_{\mathrm{pos}}$ とすると，Negative パッチ P_{neg} からランダムに選択する確率は $\bar{p}_{neg} = 1 - \bar{p}_{\mathrm{pos}}$ となる．グリッドの上から i 番目，左から k 番目のパッチ

図 3.1　グリッドサイズ 4×4 のパッチワーク画像例.

画像を $p_{i,k}$ とするとき，パッチワーク画像 $\tilde{I}_{\mathrm{pw}}$ は

$$\begin{pmatrix} p_{1,1} & \cdots & p_{1,k} & \cdots & p_{1,W_{\mathrm{g}}} \\ \vdots & \ddots & & & \vdots \\ p_{i,1} & & p_{i,k} & & p_{i,W_{\mathrm{g}}} \\ \vdots & & & \ddots & \vdots \\ p_{H_{\mathrm{g}},1} & \cdots & p_{H_{\mathrm{g}},k} & \cdots & p_{H_{\mathrm{g}},W_{\mathrm{g}}} \end{pmatrix} \tag{3.7}$$

を繋ぎ合わせたものとなる．グリッドサイズ 4×4 でパッチを繋ぎ合わせたときの変数の対応を図 3.1 に示す． グリッドの上から i_{g} 番目，左から k_{g} 番目のパッチ画像が Positive パッチであり，j 番目の物体領域 $\tilde{b}_j$ に対応するバウンディングボックス $\bar{b}_j = \{\bar{x}_j, \bar{y}_j, \bar{w}_j, \bar{h}_j\}$,

$\bar{b}_j \in B_{\mathrm{pw}}$ は

$$\bar{x}_j = k_{\mathrm{g}} W_{\mathrm{p}} + \tilde{x}_j \tag{3.8}$$

$$\bar{y}_j = i_{\mathrm{g}} H_{\mathrm{p}} + \tilde{y}_j \tag{3.9}$$

$$\bar{w}_j = \tilde{w}_j \tag{3.10}$$

$$\bar{h}_j = \tilde{h}_j \tag{3.11}$$

となる．

作成したパッチワーク画像 I_{pw} とバウンディングボックス B_{pw} を新しいデータセットとしてモデルの学習に用いる．

3.2 Few-shot モデル

少しの画像のみのデータセットから学習するという前提があるため，Few-shot Learning で用いられるニューラル物体検出モデルを使用する．農業環境から農作物を検出する場合，想定される検出対象の物体は 1 種類の農作物と考えられる．そのため，1 つのカテゴリの物体のみを抽出するためのモデルを参考にしたアーキテクチャ[15] を用いる．物体を検出する場面を表す画像をターゲット画像と検出対象物体を表す例画像をクエリ画像としてモデルに入力し，クエリ画像が表す物体のみを検出結果として出力する．このモデルのことを Few-shot モデルと呼ぶ．ターゲット画像は Faster R-CNN の入力と同じである．モデルの概要を図 3.2 に示す．

$H_{\mathrm{T}} \times W_{\mathrm{T}}$ のターゲット画像を $T \in \mathbb{R}^{3 \times H_{\mathrm{T}} \times W_{\mathrm{T}}}$ とし，$H_{\mathrm{Q}} \times W_{\mathrm{Q}}$ のクエリ画像を $Q \in \mathbb{R}^{3 \times H_{\mathrm{Q}} \times W_{\mathrm{Q}}}$ とする．はじめに，$S_{\mathrm{rs}} \times S_{\mathrm{rs}}$ にターゲット画像 T とクエリ画像 Q にリサイズし，1 つの特徴抽出器を用いてチャンネル数 C_{b}，サイズ $S_{\mathrm{b}} \times S_{\mathrm{b}}$ の特徴量 $\phi(T) \in \mathbb{R}^{C_{\mathrm{b}} \times S_{\mathrm{b}} \times S_{\mathrm{b}}}$ と $\phi(Q) \in \mathbb{R}^{C_{\mathrm{b}} \times S_{\mathrm{b}} \times S_{\mathrm{b}}}$ を抽出する．そして，Co-attention フェーズとして，抽出した特徴量の相互に注意をかけるために Non-local 操作 [21] をする．Non-local ブロックを用いて $\phi(Q)$ から $\phi(T)$ への注意のための Non-local 操作を行なった特徴量を $\psi(T;Q) \in \mathbb{R}^{C_{\mathrm{b}} \times S_{\mathrm{b}} \times S_{\mathrm{b}}}$ とする．また，同様の操作を $\phi(Q)$ から $\phi(T)$ への操作を行なった特徴量を $\psi(Q;T) \in \mathbb{R}^{C_{\mathrm{b}} \times S_{\mathrm{b}} \times S_{\mathrm{b}}}$ とする．Co-attention フェーズの最後に Non-local 特徴量を用いて拡張した特徴量 $F(T)$ と

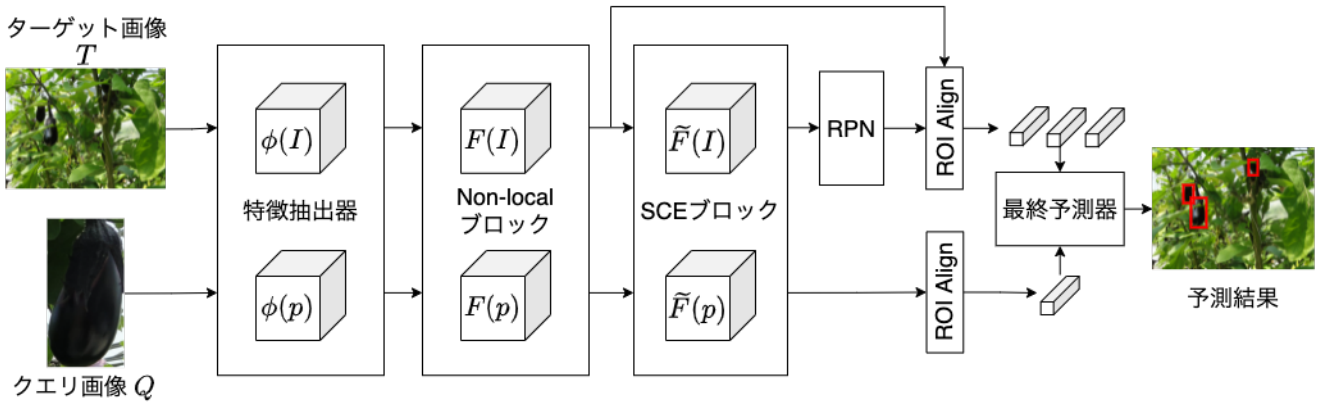

図 3.2　モデルの概要.

$F(Q)$ は

$$F(T) = \phi(T) \oplus \psi(T;Q) \in \mathbb{R}^{C_\mathrm{b} \times S_\mathrm{b} \times S_\mathrm{b}} \tag{3.12}$$

$$F(Q) = \phi(Q) \oplus \psi(Q;T) \in \mathbb{R}^{C_\mathrm{b} \times S_\mathrm{b} \times S_\mathrm{b}} \tag{3.13}$$

となる．演算子 $\oplus$ は特徴抽出器による特徴量 $\phi(\cdot)$ と non-local 特徴量 $\psi(\cdot)$ の要素ごとの足し合わせを表す.

次に Co-excitation フェーズとして，Squeeze-and-co-excitation(SCE)[22] を注意機構として用いてさらに注意をかける．“Squeeze” ステップでは，クエリ画像の拡張された特徴量 $F(Q)$ を SCE ブロックによって変換することで注意に用いる重みベクトル w を作成する．SCE ブロックによる変換では，最初に $F(Q)$ を Global Average Pooling(GAP) によって変換する．GAP によって変換された特徴量を $Z^\mathrm{a} = [z_1^\mathrm{a}, z_2^\mathrm{a}, ..., z_{C_\mathrm{b}}^\mathrm{a}]^\top, z_c^\mathrm{a} \in \mathbb{R}^{S_\mathrm{b} \times S_\mathrm{b}}$ とし，z_c^a を Z の c 番目の特徴量，$u_{c,i,j} \in F(Q)$ を $F(Q)$ の c 番目特徴量の座標 (i, j) の要素とすると，

$$z_c^\mathrm{a} = \frac{1}{S_\mathrm{b} \times S_\mathrm{b}} \sum_i^{S_\mathrm{b}} \sum_j^{S_\mathrm{b}} u_{c,i,j} \tag{3.14}$$

となる．特徴量 Z^a を 2 層のマルチレイヤーパーセプトロン (MLP)$\mathcal{M}_\mathrm{s}$ によって特徴量 $\tilde{Z}^\mathrm{a} \in \mathbb{R}^{C_\mathrm{b}}$ へ変換する．また，$F(Q)$ を Global Max Pooling(GMP) によって変換する．GMP によって変換された特徴量を $Z^\mathrm{m} = [z_1^\mathrm{m}, z_2^\mathrm{m}, ..., z_{C_\mathrm{b}}^\mathrm{m}]^\top$，$z_c^\mathrm{m} \in \mathbb{R}^{S_\mathrm{b} \times S_\mathrm{b}}$ とし，$z_{c,i,j}^\mathrm{m}$ を

Z の c 番目の特徴量の座標 (i, j) の要素とすると

$$z_c^{\mathrm{m}} = \max_{i=1,\ldots,S_{\mathrm{b}},j=1,\ldots S_{\mathrm{b}}} (u_{c,i,j}) \tag{3.15}$$

となる．“Excitation” ステップでは特徴量 Z^{m} を同様の MLP$\mathcal{M}_{\mathrm{s}}$ によって特徴量 $\tilde{Z}^{\mathrm{m}} \in \mathbb{R}^{C_{\mathrm{b}}}$ へ変換する．変換した特徴量 $\tilde{Z}^{\mathrm{a}}$ と $\tilde{Z}^{\mathrm{m}}$ を

$$w = \varsigma(\tilde{Z}^{\mathrm{a}} + \tilde{Z}^{\mathrm{m}}) \tag{3.16}$$

とすることで，$F(Q)$ より生成した重みベクトル $w \in \mathbb{R}^{C_{\mathrm{b}}}$ とする．ς はシグモイド関数を表す．最後に重みベクトル w を $F(T)$ と $F(Q)$ に

$$\tilde{F}(T) = w \odot F(T) \tag{3.17}$$

$$\tilde{F}(Q) = w \odot F(Q) \tag{3.18}$$

とすることで，$F(Q)$ から $F(T)$ への注意をかけた特徴量 $\tilde{F}(T) \in \mathbb{R}^{C_{\mathrm{b}} \times S_{\mathrm{b}} \times S_{\mathrm{b}}}$ と $F(Q)$ から $F(Q)$ への自己注意をかけた特徴量 $\tilde{F}(Q) \in \mathbb{R}^{C_{\mathrm{b}} \times S_{\mathrm{b}} \times S_{\mathrm{b}}}$ を作成する．演算子 $\odot$ は $F(T)$ と $F(Q)$ のそれぞれの特徴量のに対する w とのチャンネルごとのアダマール積を表す．

2 ステージ目では，RPN を用いてターゲット画像の拡張された特徴量 $F(T)$ から N_{r} 個の物体らしい領域候補 $R \in \mathbb{R}^{N_{\mathrm{r}} \times 4}$ を生成する．そして，RoIAlign を用いて R のそれぞれの領域に対応する特徴量 $F(T)$ 内の部分特徴量を固定サイズ S_{r} で抽出し，$\tilde{R}_{\mathrm{r}}(T) \in \mathbb{R}^{N_{\mathrm{r}} \times C_{\mathrm{b}} \times S_{\mathrm{r}} \times S_{\mathrm{r}}}$ とする．また，$\tilde{F}(Q)$ を $r \in \mathbb{R}^{C_{\mathrm{b}} \times S_{\mathrm{b}} \times S_{\mathrm{b}}}$，$r \in \tilde{R}_{\mathrm{r}}(T)$ と同様の固定長サイズの特徴量 $\tilde{R}_{\mathrm{r}}(Q) \in \mathbb{R}^{1 \times C_{\mathrm{b}} \times S_{\mathrm{r}} \times S_{\mathrm{b}}}$ に変換する．予測のための操作では，それぞれの候補領域の特徴量を平坦化した $r_{\mathrm{T}} \in \mathbb{R}^{C_{\mathrm{b}} S_{\mathrm{r}} S_{\mathrm{r}}}$，$r \in \tilde{R}_{\mathrm{r}}(T)$ と $\tilde{R}_{\mathrm{r}}(Q)$ を平坦化した $r_{\mathrm{Q}} \in \mathbb{R}^{C_{\mathrm{b}} S_{\mathrm{r}} S_{\mathrm{r}}}$ を結合した特徴量 $x = [r_{\mathrm{T}}^{\top}; r_{\mathrm{Q}}^{\top}] \in \mathbb{R}^{2C_{\mathrm{b}} S_{\mathrm{r}} S_{\mathrm{r}}}$ を作成する．特徴量 R 全体にこの操作を行なった時の特徴量 $X \in \mathbb{R}^{N_{\mathrm{r}} \times 2C_{\mathrm{b}} S_{\mathrm{r}} S_{\mathrm{r}}}$ とする．最後に，X を MLP$\mathcal{M}_{\mathrm{f}}$ によって特徴量 $\tilde{X} \in \mathbb{R}^{N_{\mathrm{r}} \times 2C_{\mathrm{b}} S_{\mathrm{r}} S_{\mathrm{r}}}$ に変換し，最終予測器で $\tilde{X}$ から N_{f} 個の物体の位置 $R_{\mathrm{f}} \in \mathbb{R}^{N_{\mathrm{f}} \times 4}$ とクラス $L_{\mathrm{f}} \in \mathbb{R}^{N_{\mathrm{f}} \times 2}$ を予測する．

RPN の最適化には Smooth L1 誤差 $L_{\mathrm{RPN\_score}}$ と 2 値クロスエントロピー $L_{\mathrm{RPN\_cls}}$ を

用いる．予測値を x，正解の値を y とすると Smooth L1 誤差 L_{smooth} は

$$L_{\mathrm{smooth}}(x,y) = \begin{cases} 0.5(x-y)^2 & if|x-y|<1 \\ |x-y|-0.5 & \text{otherwise} \end{cases} \tag{3.19}$$

と表される．予測器の最適化にも Smooth L1 誤差 L_{score} と 2 値クロスエントロピー L_{cls} を用いる．また，物体領域のスコアと背景領域のスコアの差をつけるために，Margin-based Ranking 誤差 L_{MR} を用いる．Margin-based Ranking 誤差は物体領域と背景領域スコアの差が大きく，物体領域同士のスコアの差が小さくなるように学習するための誤差である．予測スコアを X，正解クラス Y とし，x_i を X の i 番目の予測スコア，y_i を Y の i 番目の正解クラスとすると，Margin-based Ranking 誤差 L_{MR} は

$$L_{\mathrm{MR}}(X,Y) = \sum_{i=1}^{R_{\mathrm{f}}} l_{\mathrm{MR}i} + \Delta_i \tag{3.20}$$

$$l_{\mathrm{MR}i} = y_i \times \max(m^+ - x_i, 0) + (1-y_i) \times \max(x_i - m^-, 0) \tag{3.21}$$

$$\Delta_i = \sum_{j=i+1}^{R_{\mathrm{f}}} \delta_{i,j} \tag{3.22}$$

$$\delta_{i,j} = \begin{cases} \max(|x_i - x_j| - m^-, 0) & y_i = y_j \\ \max(m^+ - |x_i - x_j|,) & \text{otherwise} \end{cases} \tag{3.23}$$

$$\max(a,b) = \begin{cases} b & a \le b \\ a & \text{otherwise} \end{cases} \tag{3.24}$$

と表される．$m^+ = 0.7$，$m^- = 0.3$ とする．全体の最適化のための誤差は

$$L = L_{\mathrm{RPN\_score}} + L_{\mathrm{RPN\_cls}} + L_{\mathrm{score}} + L_{\mathrm{cls}} + \lambda L_{\mathrm{MR}} \tag{3.25}$$

となる．λ は Margin-based Ranking 誤差に対する係数を表す．

第 4 章

実験方法

4.1 農業データセット

農作物としてナスを検出対象として物体検出を行うためのデータセットとし，農業データセットとする．このデータセットで使用する画像は高知県農業技術センターの温室内のナス畑から人が撮影したものである．画像は少なくとも 1 つのナスを含み，撮影場所や向きはランダムである．また，撮影した時間，日照条件，移り方は考慮しない．画像の大きさは縦 2592 ピクセルと横 3456 ピクセルである．ナスの位置情報とクラスラベルのアノテーションは人の手作業によって作成する．データセットには 98 枚の画像が含まれる．

データセットの画像に含まれるナスの写り方は考慮していないため，近くにあるナスから遠くにあるナスまで含まれており，大きさに違いがある．その大きさを元に複数の難易度をもつデータセット Easy，Normal，Hard を作成する．大きい物体のみが含まれるデータセットから大きい物体から小さい物体の全てが含まれるデータセットを作成し，Easy，Normal，Hard の順で難易度が高くなると考える．作成したデータセットとそのデータセットに含まれる最小の物体のサイズ，物体の数を表 4.1 に示す．

表 4.1 各データセットに含まれるアノテーションの数と検出対象となる物体のサイズ．

データセット	対象物体サイズ	アノテーション数
Easy	192 × 192 以上	323
Normal	96 × 96 以上	685
Hard	1 × 1 以上 (全ての物体)	1155

表 4.2　Few-shot 設定を導入した各学習用データセットに含まれるアノテーションの数と検出対象となる物体のサイズ.

データセット	対象物体サイズ	アノテーション数
Easy	192 × 192 以上	16
Normal	96 × 96 以上	41
Hard	1 × 1 以上 (全ての物体)	84

Few-shot 設定を取り入れ，少ない画像のみで学習することを目的としたデータセットにする時，全ての画像から 5 枚画像を抽出し，Few-shot 設定を取り入れた学習データセットとする．また，残りの 93 枚の画像は Few-shot 設定を取り入れたテストデータセットとする．学習用画像 5 枚を抽出するとき，ランダムに抽出する．基本的に，全てのデータセットに Few-shot 設定を導入する．Few-shot 設定を導入した学習データセットに含まれる物体の数を表 4.2 に示す.

Few-annotation 設定を取り入れ少ない画像のみで学習することを目的としたデータセットにする時，全てのアノテーションから 5 つを抽出し，Few-annotation 設定を取り入れた学習データセットとする．また，残りのアノテーションは削除する．学習用アノテーションを 5 つ抽出する時，アノテーションのコストなどを考慮し，大きく葉による隠れのないアノテーションがしやすいナスを恣意的に抽出する.

各難易度の学習データセットに含まれる物体領域を示すアノテーションの数を図 4.1 に示す． 各難易度における学習データセットに含まれる物体領域を示すアノテーションの 1 辺の平均サイズごとの数を図 4.2 に示す． データセット Hard における，学習用データセット，テスト用データセット，Few-annotation 設定を導入した学習データセットに含まれる物体領域を示すアノテーションの 1 辺の平均サイズごとの数を図 4.3 に示す.

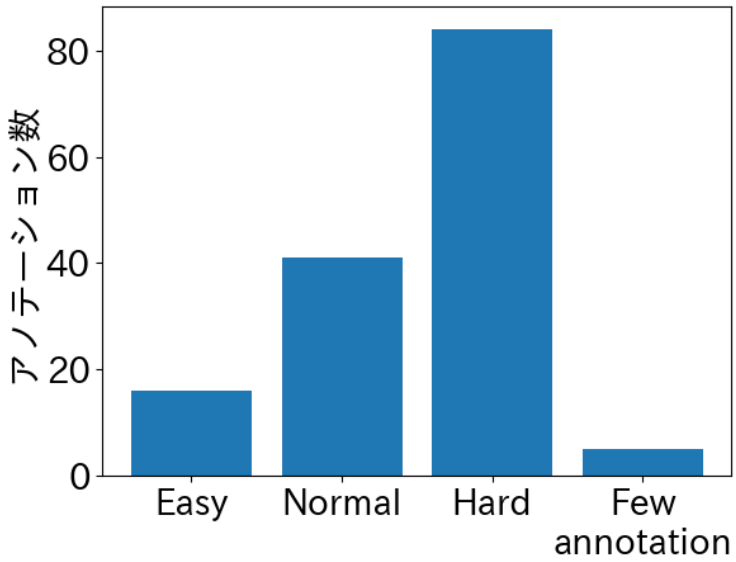

図 4.1　各データセットにおけるアノテーション数．Easy，Normal，Hard はデータセット Easy，Normal，Hard を示し，Few-annotation は Few-annotation 設定を導入したデータセットを示す．

4.2　モデル学習

Few-shot モデルの事前学習では，物体検出用のデータセットの MS-COCO `train2017`[23] を用いる．クエリ画像として用いる画像は事前学習済みの Mask R-CNN で検出が可能である物体の画像のみを候補とする．ターゲット画像に含まれるカテゴリからランダムに 1 つのカテゴリを選択し，クエリ画像の候補の集合から同カテゴリのクエリ画像をランダムに 1 枚選択することで，ターゲット画像とクエリ画像のペアを作成する．このペアはあらかじめ決めておき，全ての学習エポックにおいて同じ組み合わせで学習を行う．Few-shot モデルの特徴抽出器には ResNet50 と Feature Pyramid Network を用いる．この特徴抽出器の上位 3 層を学習し，下位 2 層は学習を行わない．Few-shot モデルの事前学習で使用するハイパーパラメータを表 4.3 に示す．入力画像を [0, 1] に正規化した後に，データセットの画素平均 RGB 値 (0.485, 0.456, 0.406) と画素標準偏差 RGB 値 (0.229, 0.224, 0.225) の値で正規化する．入力画像は縦幅と横幅を (512, 512) にリサイズして，モデルの入力値とする．学習には GPU である NVIDIA TESLA V100 を 8 枚を使用し，並行

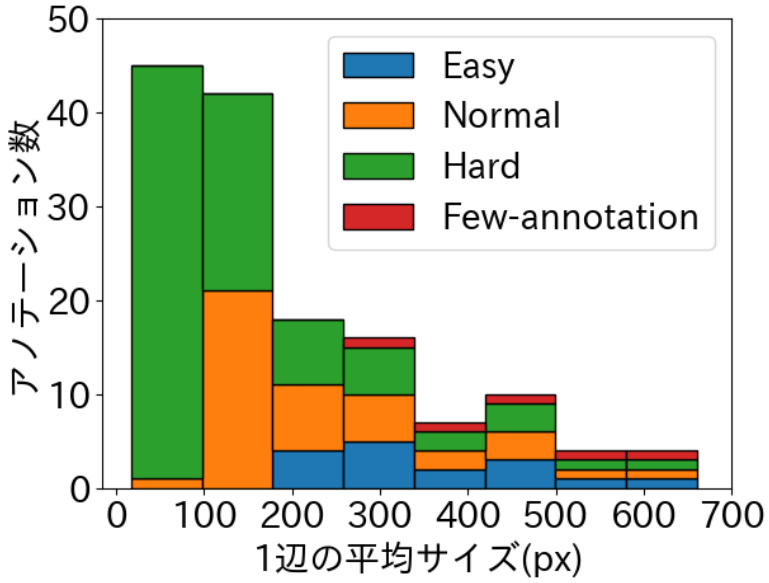

図 4.2　各データセットに含まれるアノテーションの 1 辺の平均サイズごとの数．Easy はデータセット Easy，Normal はデータセット Normal，Hard はデータセット Hard に含まれるアノテーションを示す．Few-annotation は Few-annotation 設定を導入したデータセットのことを示す．

表 4.3　Few-shot モデルの事前学習に使用するハイパーパラメータ

Name	Value
エポック	50
バッチサイズ	32
学習率	10^{-4}
momentum	0.9
weight decay	10^{-3}

に学習する．この事前学習用データセットで学習したモデルを事前学習済みモデルとする．

農業データセットでの事前学習済みモデルの学習では Few-shot 設定と Few-annotation 設定を考慮し，5 枚の画像かつ 5 つのアノテーションのみを使用して学習するため，提案手法で作成したデータセットを用いてモデルを学習する．5 つのアノテーションは恣意的に選択した大きく葉による隠れのないナスを 5 つ使用する．作成データセットに含まれる画像のサイズは縦幅と横幅を (512, 512) とし，5 枚の各画像のグリッドのサイズは (4,4), (4,4), (8,8), (8,8), (16,16) とする．Positive パッチの作成にはデータセットに含まれる 5 つのア

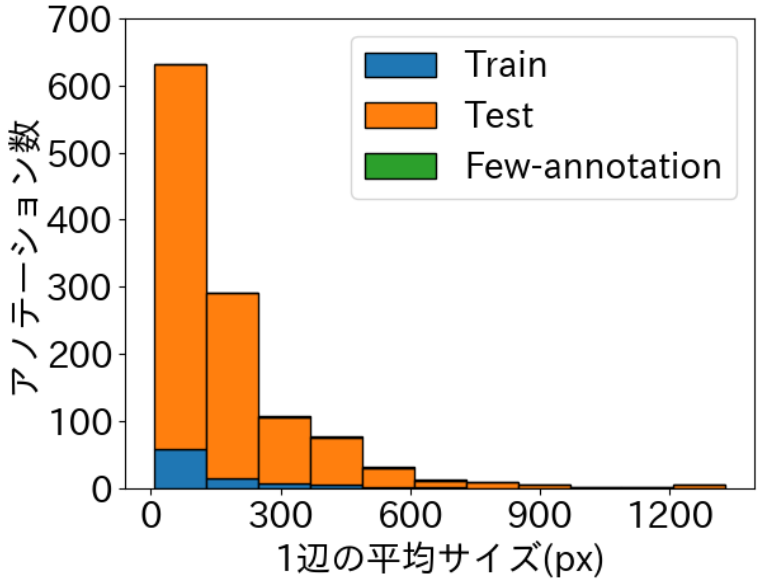

図 4.3　データセット Hard における，学習用，テスト用，Few-shot 設定を導入した学習用に含まれるアノテーションの 1 辺の平均サイズごとの数．Train は学習用データセットを示し，Test はテスト用データセットを示す．Few-annotation は Few-annotation 設定を導入したデータセットを示す．

ノテーションを使用する．Negative 候補パッチの作成にはランダムに抽出した縦幅と横幅が (128, 128) の部分画像を使用し，それを 2048 枚作成する．Negative パッチの作成では，Negative 候補パッチと Positive パッチを k-means によって 8 クラスタに分類し，Positive パッチと同クラスタに分類された Negative 候補パッチを Negative 候補パッチから除去する．パッチワーク画像の作成で Positive パッチと Negative パッチを選択するが，各画像における Positive パッチを選択する確率を 0.5, 0.5, 0.3, 0.3, 0.2 とする．パッチのリサイズにはバイリニア補完による方法を使用する．クエリ画像は 5 つのアノテーションからランダムに選択する．クエリ画像とターゲット画像のペアはあらかじめ決めておき，全ての学習エポックにおいて同じ組み合わせで学習を行う．パッチワーク拡張によって作成したデータセットによる Few-shot モデルの学習に使用するハイパーパラメータを表 4.4 に示す． 入力画像を [0,1] に正規化した後に、データセットの画素平均 RGB 値 (0.452, 0.430, 0.398) と画素標準偏差 RGB 値 (0.281, 0.271, 0.282) の値で正規化する．バッチサイズ 1 として，

表 4.4　Few-shot モデルの Finetuning に使用するハイパーパラメータ

Name	Value
エポック	500
バッチサイズ	5(1)
最適化関数	Adam
学習率	10^{-6}
momentum	0.9
weight decay	10^{-3}
Margin-based Ranking 誤差 lambda	3

表 4.5　実験条件一覧．①は Few-shot 設定．②は Few-annotation 設定．③は SSD のデータ拡張．④はパッチワーク拡張．

モデル	ケース名	①	②	③	④
Faster R-CNN	case1	✓			
	case2	✓		✓	
	case3	✓	✓	✓	
Few-shot モデル	case4	✓			
	case5	✓	✓		
	提案手法	✓	✓		✓

gradient accumulation を用いて，5 イテレーション毎に最適化を行うことで，バッチサイズ 5 として学習する．

事前学習済み Few-shot モデルのパッチワーク拡張データセットでの学習結果をその他のモデルと比較する．実験条件一覧を表 4.5 に示す． case1 は Few-shot 設定や Few-annotation 設定を考慮していない一般的なニューラル物体検出モデルとして事前学習済みの Faster R-CNN を Few-shot 設定を取り入れた農業データセットでの学習結果とする．case2 は case1 の Faster R-CNN の学習に物体検出モデル SSD で用いられるデータ拡張手法 [6] を追加した実験とする．case4 は事前学習済み Few-shot モデルを Few-shot 設定を取り入れた農業データセットで学習した結果とする．case3 と case5 は事前学習済みの Faster R-CNN と

Few-shot モデルを Few-shot 設定と Few-annotation 設定を取り入れた農業データセットでデータセット拡張をせずに学習した結果とする．提案手法は事前学習済みの Few-shot モデルをパッチワーク拡張したデータセットで学習した結果とし，マージンの最大値の各要素に 300 を用いる．また、事前学習済み Few-shot モデルのパッチワーク拡張したデータセットを用いた学習のみの比較として、マージンの最大値の各要素を 0 から 500 に 100 ずつ変化させたときの結果を比較する比較に使用する Faster R-CNN は特徴抽出器として ResNet50 と Feature Pyramid Network を用い，は MS-COCO `train2017` によって事前学習したものを使用する．また，Faster R-CNN は torchvision により実装されたものを使用する．

比較に使用する指標は Average Precision 50(AP50) と Average Precision 75(AP75) とする．ターゲット画像とクエリ画像のペアの生成やパッチワーク画像の生成のランダム性を考慮して，500 エポックの学習を 10 試行行う．そのため，最終的な比較指標としては，全試行における各指標の平均値とする．この全ての実験を全ての難易度のデータセットで試行する．

第 5 章

結果と考察

5.1 結果

全ての難易度のデータセット，全ての実験条件における 500 エポック時の 10 試行平均 AP50, AP70 を表 5.1 に示す． また，提案手法と各比較手法の各難易度における 10 試行平均の AP50, AP75 の学習曲線を図 5.1, 5.2 に示す． データセット Easy の 500 エポックにおいて，提案手法の AP50 は Few-shot 設定が導入されたデータセットを拡張して Faster R-CNN を学習した Case2 より 5.8 ポイント高く，Few-shot 設定が導入されたデータセットで Few-shot モデルを学習した Case4 より，0.5 ポイント低い．データセット Hard の 500 エポックにおいて，提案手法の AP50 は Case2 より 15.0 ポイント高く，Case4 より，4.8

表 5.1 全ての実験条件における結果一覧．結果は 500 エポックでの指標 AP50 と AP75.

ケース名	Easy		Normal		Hard		アノテーション数		
	AP50	AP75	AP50	AP75	AP50	AP75	Easy	Normal	Hard
case1	24.0	14.3	14.7	7.7	10.7	5.0	16	41	84
case2	45.8	28.8	22.1	13.1	11.1	5.0	16	41	84
case3	3.1	2.2	1.5	1.2	1.0	0.7	5	5	5
case4	52.0	34.8	45.0	22.6	30.8	14.0	16	41	84
case5	0.7	0.3	0.4	0.2	0.3	0.1	5	5	5
提案手法	51.6	30.8	40.9	20.9	26.0	13.0	5	5	5

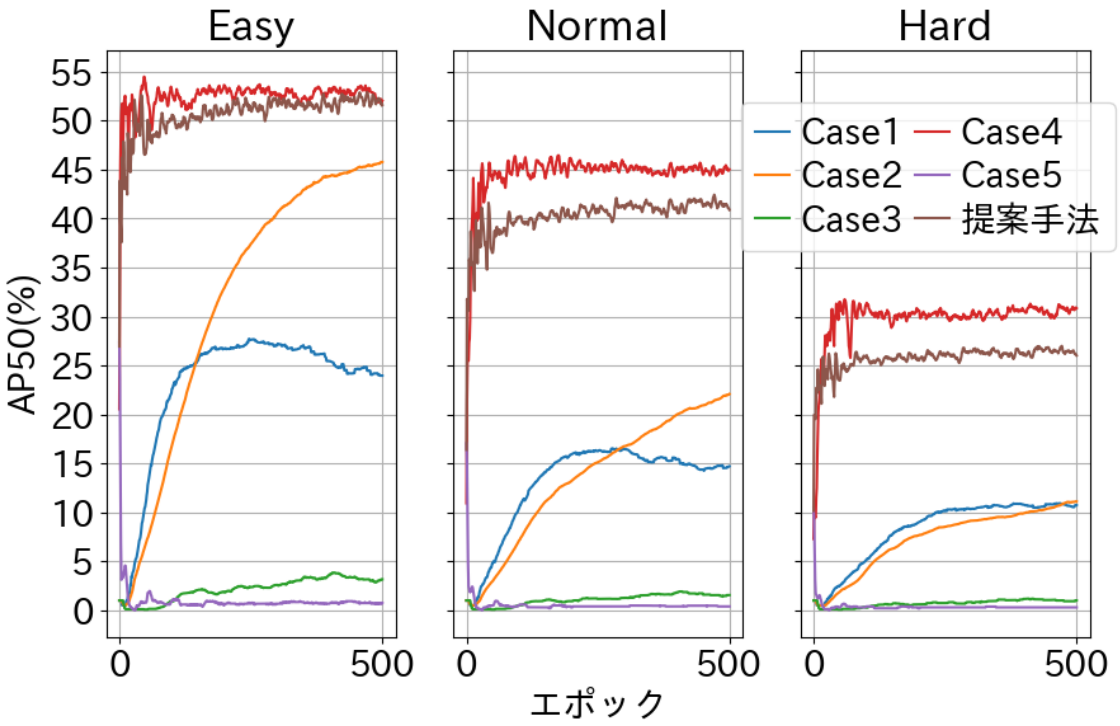

図 5.1　各難易度のデータセットにおける 500 エポックの学習による AP50

表 5.2　各手法に要する平均時間

名前	1 エポック	1 イテレーション	推論
Faster R-CNN データ拡張なし	0.880 s	0.069 s	0.046 s
Faster R-CNN データ拡張あり	2.446 s	0.742 s	0.051 s
Few-shot モデル	1.355 s	0.109 s	0.061 s

ポイント低い.

各手法での 500 エポック 10 試行したときの 1 エポックと 1 イテレーションにおけるモデル学習にかかる時間と推論時間を表 5.2 に示す. Few-shot モデルが 1 エポック学習するのに必要な時間はデータ拡張なしの Faster R-CNN の 1 エポックの学習時間より 0.5 秒多く, データ拡張ありの Faster R-CNN の 1 エポックの学習時間より 1.1 秒少ない.

提案手法において, マージンを変化させたときの 500 エポック時の 10 試行平均 AP50,AP75 を表 5.3 に示す. 提案手法においてマージンを変化させたときの AP50, AP75 の学習曲線を図 5.3, 5.4 に示す. データセット Easy の 500 エポックにおいて, マー

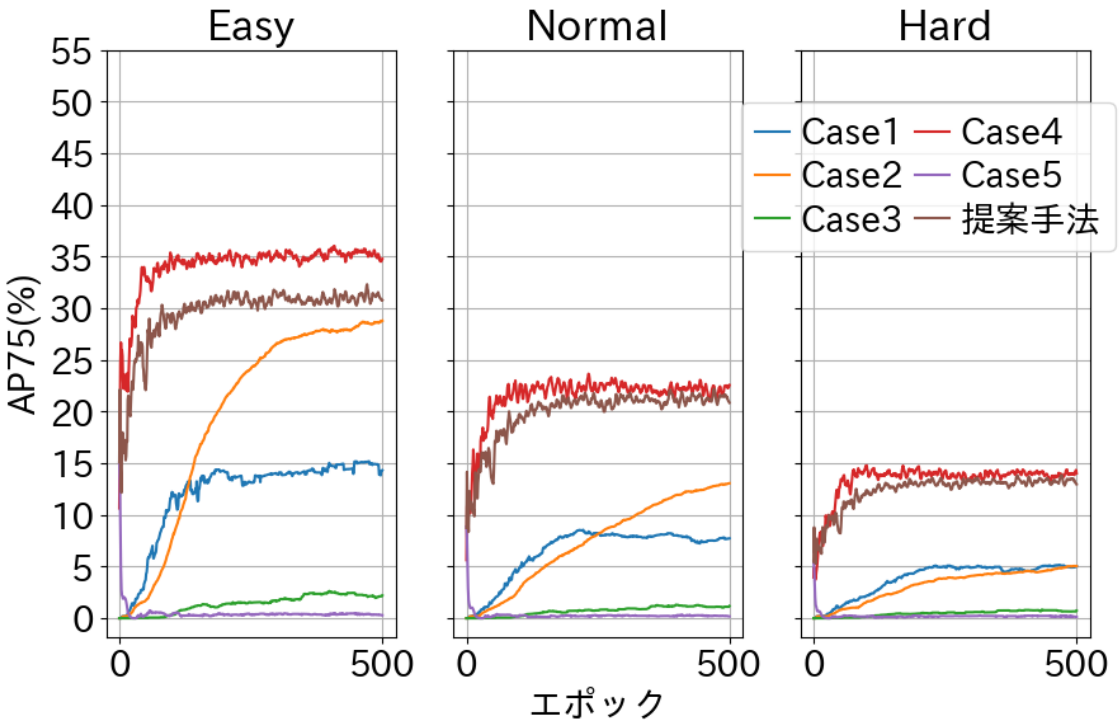

図 5.2　各難易度のデータセットにおける 500 エポックの学習による AP75

ジンの最大値を 100 とした場合の AP50 はマージンの最大値を 500 とした場合より 5.9 ポイント高く，マージンを取らなかった場合より 9.7 ポイント高い．データセット Hard の 500 エポックにおいて，マージンを 500 取った場合の AP50 はマージンを 100 取った場合より 3.0 ポイント高く，マージンを取らなかった場合より 9.4 ポイント高い．

Few-shot モデルを Few-shot 設定をと取り入れたデータセットで学習させた Case4 の 500 エポック時の予測結果を 5.5(b) に示す．また，Few-shot モデルを Few-shot かつ Few-annotation 設定を取り入れたデータセットでパッチワーク画像を作成して学習した場合の 0 エポック時と 500 エポック時の予測結果を図 5.5(a),5.5(c) に示す．

5.2　考察

さまざまな観点から今回の結果に対して考察を行う．

表 5.3 提案手法のマージンを変化させたときの結果一覧．結果は 500 エポックでの指標 AP50 と AP75.

マージン	Easy		Normal		Hard	
	AP50	AP75	AP50	AP75	AP50	AP75
0	45.9	27.1	28.1	15.4	17.2	9.5
100	55.6	34.0	38.9	21.6	23.7	13.3
200	53.8	32.2	41.9	22.2	26.1	13.7
300	51.6	30.8	40.9	20.9	26.0	13.0
400	51.5	30.8	40.9	21.5	26.8	13.5
500	49.6	27.7	39.4	19.4	26.6	12.3

5.2.1 Few-annotation 設定を導入してパッチワーク拡張を導入しなかった場合

Few-shot 設定と Few-annotation 設定の両方を導入したデータセットを用いてパッチワーク拡張なしで Faster R-CNN と Few-shot モデルを学習した場合，AP50 と AP75 が数十エポック学習すると同時に減少していることが図 5.1，5.2 からわかり，最後の 500 エポック時に提案手法より AP50 と AP75 が低いことが表 5.1 よりわかることから，これらの場合では学習できていないことがわかる．また，Few-shot 設定と Few-annotation 設定を導入したデータセットを用いてパッチワーク拡張ありで Few-shot モデルを学習した場合，AP50 と AP75 が数十エポックの学習と同時に増加していることが図 5.1，5.2 よりわかることから，提案手法では学習ができていることがわかる．これは，アノテーションのある物体より多くのアノテーションのない物体が画像内に含まれており，アノテーションのない物体を背景として学習してしまっているため，アノテーションのある物体に対しても背景として予測していると考えられる．また，学習データに含まれるアノテーションのある物体のみを検出対象の物体として過剰に学習し，学習した物体以外の物体は背景として判定するように過学

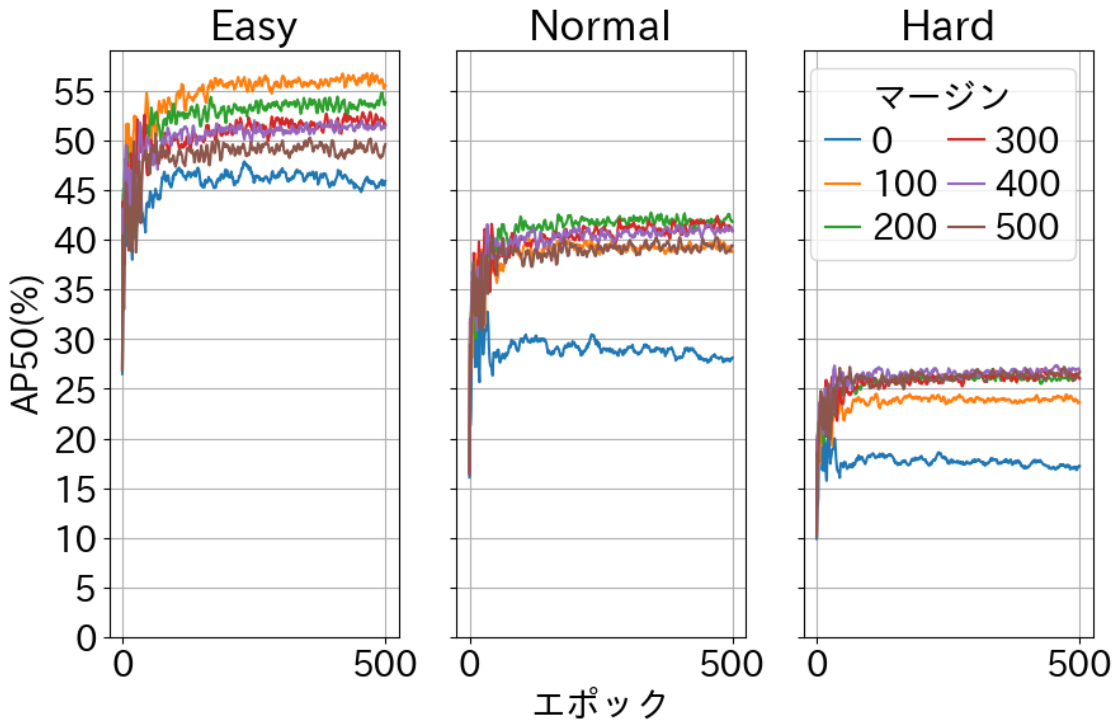

図 5.3　各難易度のデータセットにおけるマージンを変化させた 500 エポックの学習による AP50

習している可能性も考えられる．提案手法においてはアノテーションのない物体が含まれにくくなっているため，背景として物体が学習されておらず，テストデータにおいても物体を物体として予測できていることが考えられる．

5.2.2　提案手法のマージンの最大値について

Few-shot かつ Few-annotation 設定を取り入れたデータセットを用いてパッチワーク画像を作成するときのマージンの最大値を大きくすると，物体領域の周辺領域を大きく取ることになるため，物体のスケールを変化させることができ，小さな物体として学習できる．これによって小さな物体の検出の能力向上が考えられるが，同時にアノテーションのない物体がノイズとして学習データに含まれる可能性も出てくる．検出対象が大きな物体のみであるデータセット Easy では，マージンの最大値が 100 や 200 のときの AP50,AP75 の方が 400 や 500 の時の AP50,AP75 よりも大きいことが表 5.3 と図 5.3，5.4 から確認することがで

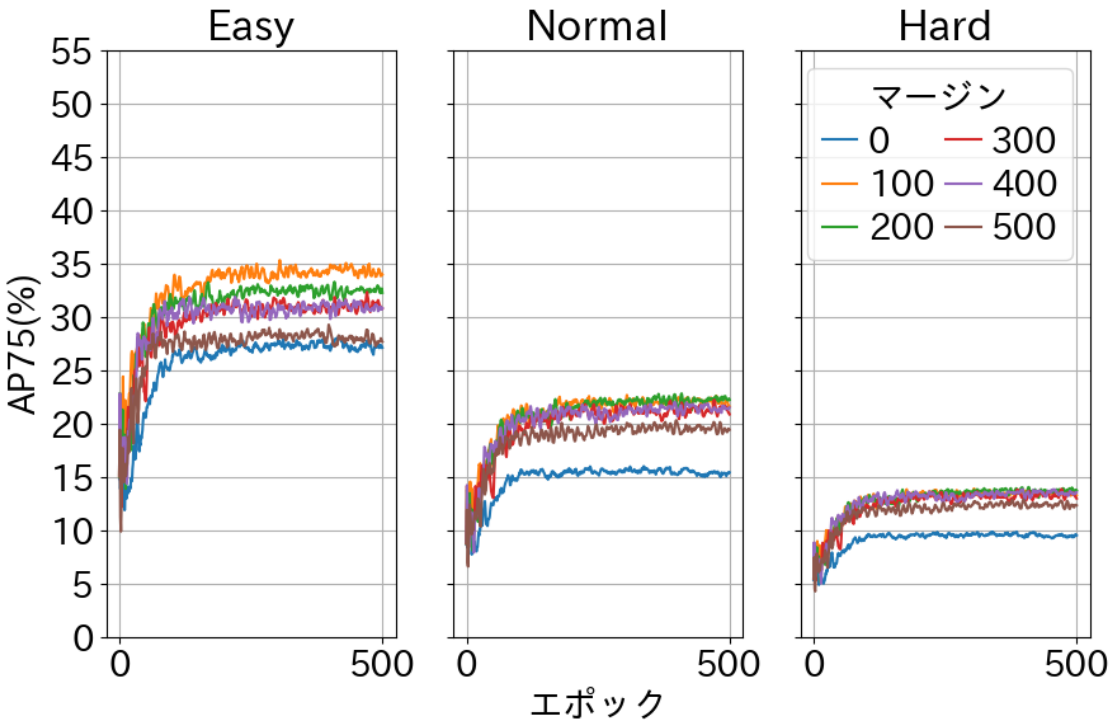

図 5.4　各難易度のデータセットにおけるマージンを変化させた 500 エポックの学習による AP75

きる．これらのことから，検出対象が大きな物体のみである場合，マージンの最大値を大きく取るときのスケールの変化による好影響よりもアノテーションのない物体をノイズとして学習してしまう悪影響の方が大きくなることが考えられる．他にも，データセット Easy には小さい物体が含まれないため，学習時に小さな物体のみを検出するように学習してしまっていることも大きな要因と考えられる．一方，検出対象が大きな物体から小さい物体の全てであるデータセット Hard では，マージンの最大値が 400 や 500 のときの AP50,AP75 の方が 100 や 200 の時の AP50,AP75 よりも大きいことが表 5.3 と図 5.3，5.4 から確認することができる．これらのことから，検出対象がさまざまな大きさの物体である場合，マージンの最大値を大きく取るときのスケール変化による好影響がアノテーションのない物体をノイズとして学習してしまう悪影響より大きくなることが考えられる．

また，物体領域の周辺領域をパッチ画像に含めない場合，小さい物体の検出が困難になっていることがデータセット Hard でモデルを学習させたときの表 5.3 の AP50 と AP75 から確認できる．データセット Easy においてもマージンを取らずにパッチワーク画像を作成し

た場合は AP50 と AP75 が低くなっていることが確認できるため，マージンを取ることによる物体のスケール変化や物体領域の表現の変化がデータの学習に良い影響を与えていると確認することができる．

5.2.3 Few-shot モデルの予測について

Few-shot モデルの予測結果について定性的に評価をする．Few-shot モデルは 0 エポック時の農業データセットによる学習を行わない場合でも，いくつかのナスを予測することが可能であると図 5.1 の 0 エポック時の AP50 と予測結果の図 5.5(a) から確認できる．Few-shot モデルの 500 エポック時の予測 (図 5.5(b), 5.5(a)) からは大きく写っているナスや葉に隠れていないナスは検出ができていることが確認できる．Few-shot モデルを Few-shot 設定のみを導入したデータセットで学習させた Case4 の方が，いくつかの小さいナスや葉に隠れたナスの隠れていない部分を予測できているが，Few-shot 設定と Few-annotation 設定を導入したデータセットで学習した提案手法は，葉による隠れや小さいナスに関しては大きいナスや葉に隠れていないナスに比べて検出ができていないことが確認できる．また，一部の予測から，葉による隠れがあるナスを検出できた場合でも，見えている部分を 1 つのナスと判定することからナスの上部と下部が別々のナスとして予測されてしまっていることが確認できる．提案手法において，これら全ての問題は物体領域としてこのような例を学習に含めていないことが原因であると考えられる．パッチワーク画像を作成する段階で擬似的に葉による隠れを作成した物体領域を学習に加えるなどしたデータでモデルを学習させることができると考えられる．さらに，背景領域として学習に含まれなかった農業器具などの物体がナスとして判定されていることが，いくつかの予測結果 (図 5.5(c) の左から 3 番目) から確認することができる．ランダムに背景領域を抽出していることから，抽出しきれていない対象外物体の学習ができていないと考えられる．人の目視作業による物体領域の抽出のように，検出対象外の物体や背景領域の抽出も人間によっていくつか抽出して学習に含めることによって，今回誤検出された物体に対しては回避が可能と考えられる．

5.2.4 各モデルの学習速度

Few-shot モデルを用いてデータセットに適用させる場合，学習に必要なエポック数が 100 エポックほどかつ 1 エポックにかかる時間が 1.3 秒ほどとなっているため，学習に必要なエポック数と 1 エポックにかかる時間においてデータセット拡張がある Faster R-CNN より小さい．そのため，Few-shot モデルを用いた方が Faster R-CNN を用いるより素早く各農業環境に最適化された検出モデルを作成することができると考えられる．

(a) Few-shot モデル 0 epoch

(b) Case4 500 epoch

(c) 提案手法 500 epoch

図 5.5　画像に描画した予測結果．黄色の矩形が予測ラベル．データセット Easy には赤色の矩形が正解ラベルとして含まれる．Normal データセットには赤色と緑色の矩形が正解ラベルとして含まれる．Hard データセットには赤色と緑色と青色の矩形が正解ラベルとして含まれる．赤色，緑色，青色の順番に小さい物体を示す．図 5.5(a) は事前学習済み Few-shot モデルを農業データセットで学習せずに予測した結果．図 5.5(c) は事前学習済み Few-shot モデルを Few-shot 設定を導入した農業データセットでパッチワーク画像を作成して学習した結果．図 5.5(c) は事前学習済み Few-shot モデルを Few-shot 設定と Few-annotation 設定を導入した農業データセットでパッチワーク画像を作成して学習した結果．

第 6 章

結論

本研究では，ニューラルネットワークによる Few-shot 検出モデルとデータセットの拡張を用いてモデルの学習に必要なデータセットの数と学習時間減らすための手法を提案し，ナス畑のナスを検出するための物体検出データセットを用いて実験を行った．実験では意図的にデータセットに含まれるアノテーションの数を減らし，減らす前のデータセットを用いたときの場合と比較をした．全ての大きさのナスを含むデータセットにおける実験では，アノテーションの数を約 90%減らしたデータセットで提案手法を学習したときの AP50 はアノテーションを減らす前の Faster R-CNN より 15.0 ポイント高く，アノテーションを減らす前の Few-shot モデルより 4.8 ポイント低いことを示した．また，最も高い AP50 を獲得するまでの学習時間は Few-shot モデルを用いた場合は 1 エポック 1.3 秒を 100 エポックに対して，データ拡張ありの Faster R-CNN は 1 エポック 2.4 秒を 500 エポック，データ拡張なしの Faster R-CNN は 1 エポック 0.8 秒を 500 エポックと計測であったため，Few-shot モデルを使用することで学習に必要な時間を減らすことができると確認できた．そして，Few-annotation 設定をデータセットに導入した場合，パッチワーク画像を作成せずに Faster R-CNN と Few-shot モデルを学習させると AP50 が 0 ポイント付近に低下したことから，学習ができないことが確認できた．Patchwork 画像を作成するときの物体周辺の画像をどれだけ含むかを変化させることによって，アノテーションのない物体が含まれるときの影響とスケール変化の影響を確認することができた．最後に，検出対象以外の物体領域や検出対象物体ににている背景領域を誤検出しているため，人の作業による背景領域の作成などで対応をする必要があると考えた．

謝辞

本研究を行うにあたって，ご指導していただいた吉田真一教授に感謝致します．このように学ぶことができたのも吉田研究室という環境があったためであると思っています．6 年間ありがとうございました．

四宮友貴助教にもまた多くのことを学ばせていただける機会をいただき感謝致します．研究を進めるにあたっての具体的な技術についての相談やデータの収集など，多くのことでお世話になったと思います．

福本昌弘教授と栗原徹教授には副査をしていただき，さらには研究発表直前に発表内容に対する助言をしていただき感謝致します．

IoP プロジェクトと高知県農業技術センターには，本研究に多大なご協力を頂きましたことを感謝します．

最後に，大学生活 6 年間の研究面，生活面において支えてくださった方に心から感謝致します．

参考文献

[1] “農業労働力に関する統計：農林水産省”. https://www.maff.go.jp/j/tokei/sihyo/data/08.html. (Accessed on 01/23/2022).

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. “Deep residual learning for image recognition”. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[3] Andreas Veit, Michael J Wilber, and Serge Belongie. “Residual networks behave like ensembles of relatively shallow networks”. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, Vol. 29. Curran Associates, Inc., 2016.

[4] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. “Feature pyramid networks for object detection.”. In *CVPR*, pp. 936–944. IEEE Computer Society, 2017.

[5] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. “You only look once: Unified, real-time object detection”. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[6] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. “Ssd: Single shot multibox detector”. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pp. 21–37, Cham, 2016. Springer International Publishing.

[7] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. “Region-based convolutional networks for accurate object detection and segmentation”. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 38, No. 1, pp. 142–158, 2016.

[8] Ross Girshick. “Fast r-cnn”. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1440–1448, 2015.

[9] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. “Faster r-cnn: Towards real-time object detection with region proposal networks”. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, Vol. 28. Curran Associates, Inc., 2015.

[10] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. “Mask r-cnn”. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2980–2988, 2017.

[11] Bingyi Kang, Zhuang Liu, Xin Wang, Fisher Yu, Jiashi Feng, and Trevor Darrell. “Few-shot object detection via feature reweighting”. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[12] Juan-Manuel Perez-Rua, Xiatian Zhu, Timothy M. Hospedales, and Tao Xiang. “Incremental few-shot object detection”. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[13] Claudio Michaelis, Ivan Ustyuzhaninov, Matthias Bethge, and Alexander S. Ecker. “One-shot instance segmentation”. *arXiv*, 2018.

[14] Qi Fan, Wei Zhuo, Chi-Keung Tang, and Yu-Wing Tai. “Few-shot object detection with attention-rpn and multi-relation detector”. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[15] Ting-I Hsieh, Yi-Chen Lo, Hwann-Tzong Chen, and Tyng-Luh Liu. “One-shot object detection with co-attention and co-excitation”. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, Vol. 32. Curran Associates, Inc., 2019.

[16] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. “mixup: Beyond empirical risk minimization”. In *International Conference on Learning*

Representations, 2018.

[17] Terrance Devries and Graham W. Taylor. “Improved regularization of convolutional neural networks with cutout”. *CoRR*, Vol. abs/1708.04552, , 2017.

[18] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. “Random erasing data augmentation”. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2020.

[19] Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D. Cubuk, Quoc V. Le, and Barret Zoph. “Simple copy-paste is a strong data augmentation method for instance segmentation”. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2918–2928, June 2021.

[20] S. Hong, S. Kang, and D. Cho. “Patch-level augmentation for object detection in aerial images”. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pp. 127–134, Los Alamitos, CA, USA, oct 2019. IEEE Computer Society.

[21] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. “Non-local neural networks”. *CVPR*, 2018.

[22] Jie Hu, Li Shen, and Gang Sun. “Squeeze-and-excitation networks”. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[23] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. “Microsoft coco: Common objects in context”. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pp. 740–755, Cham, 2014. Springer International Publishing.