

ディープラーニングを活用した画像認識に関する研究

増山 史倫^{*1)}, 印田 清秀^{*2)}, 林田 平馬^{*1)}, 木田 裕之^{*1)}

Study of Image Recognition by Deep Learning

MASUYAMA Fumihito^{*1)}, INDA Kiyohide^{*2)}, HAYASHIDA Heima^{*1)}, KIDA Hiroyuki^{*1)}

近年、画像認識性能が機械学習の手法の一つであるディープラーニングにより著しく発展している。本研究では、ディープラーニングによる画像認識技術を用いて、実際の事例として農作物である小ギクの生育状況の判別への適用を試みた。収集された画像から学習データセットを構築し、生育状況を判別する生育状況判別モデル（以下、「判別モデル」という）の学習及び識別を行い、課題や適用の可能性について検討した。

1. 緒言

画像認識とは、画像データに何が写っているか、または、その状態などを、機械やコンピュータが自動的に判別する技術であり、近年、ディープラーニングといわれる手法によりこの性能が著しく向上している。この技術を様々な分野に応用すれば、人手や手間の低減を図ることができ、作業工程などの自動化や省力化などにつながる。

当センターでは、ディープラーニングによる画像認識技術を様々な分野において活用する検討を進めている。この報告では、小ギクの生育状況の判別における事例を対象として、手法を用いる課題や今後の可能性について検討した。

奈良県の小ギクは生産量全国第2位の主要品目である。しかし、近年は、春から夏にかけて気温の高い年が多く、その影響で開花時期が不安定となり、需要期である盆時期の安定出荷が難しくなっている。安定出荷を図るため、生育状況や環境条件から収穫時期を予測することが考えられる。しかし、生育状況は、地域の気候・土壌などの環境要因や日当たりなどの局所的な環境要因によって、様々な変化するため、簡便に調査する方法が求められる。そこで、小ギクの画像を対象に生育状況の判別へのディープラーニングの適用を図った。

2. 手法

奈良県農業研究開発センターが収集した圃場における小ギクの画像から、ディープラーニング学習用データベースを作成し、判別モデルの学習を行い、検証を行う。ソフトウェアの実装は、プログラミング言語 Python で行い、ディープラーニングフレームワークとして、Keras¹⁾ の Tensorflow バックエンド²⁾を用いた。また、画像処理については、OpenCV³⁾を用いた。

2.1 画像ファイルの準備

判別モデルの学習には、奈良県農業研究開発センターにおける試験圃場にて小ギクを真上から撮影した画像を使用した。撮影した画像の例を図1に示す。画像ファイルのピクセルサイズは 3168px×4752px であった。撮影日を判別モデルにおけるクラスとすることとする。

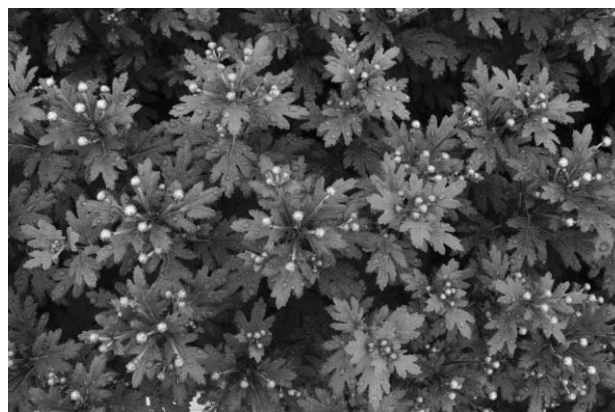


図1 撮影画像例

2.2 ディープラーニングの手法

用意できる学習データが少ないことやはじめから判別モデルを学習するには計算量が膨大になることから、既に学習されたディープラーニングモデルの一部を変更して再学習し、新しいクラスの判別についても適合できるファインチューニング⁴⁾と呼ばれる手法を用いる。今回は、InceptionV3⁵⁾の学習済みモデルを利用し、全314層のうち第250層以降の層内にある畳み込み層と全結合層の重みを再学習し、判別モデルを作成することとした。なお、入力層から249層までは、画像の特徴の一般的なことをとらえているとされるため、そのまま重みを固定することとする。

^{*1)}奈良県産業振興総合センター IoT 推進グループ

^{*2)}奈良県農業研究開発センター

2.3 画像の前処理

判別モデルの学習データベースを作成するため、元画像に対して前処理を行う。

判別モデルの学習に用いる画像のサイズは、InceptionV3の学習済みモデルの入力である 299px×299px とした。画像を過度に縮小すると認識したい特徴が失われてしまうため、ある程度の特徴が認識できる大きさになるよう次のとおり OpenCV により画像処理を行った。元画像を 1440px×1920px に縮小し、任意の 70 か所の位置において 299px×299px サイズで元画像を切り抜いた。この際、地面などの対象物が写っていない画像を削除するため、HSV 色空間で彩度の平均が 30 に満たない画像は除外する。

2.4 データ拡張

過学習を避けるため、前処理後画像に移動、変形、反転、回転、拡大・縮小、明度などに変換処理を加えてデータの拡張を行う。Keras では、ImageDataGenerator⁶⁾によりデータ拡張が可能であり、それぞれのパラメータを表 1 のとおりとした。

表 1 データ拡張のパラメータ

パラメータ	説明	
rescale	画素値のリスケーリング係数 (正規化)	1.0 / 255
rotation_range	画像を回転させる角度	20
width_shift_range	水平にシフトする画像横幅に対する割合	0.1
height_shift_range	水平にシフトする画像縦に対する割合	0.1
horizontal_flip	水平方向反転	True
vertical_flip	垂直方向反転	True
zoom_range	拡大縮小範囲	0.2
shear_range	シアア強度	5
brightness_range	明度の範囲	[0.3, 1.0]

2.5 学習

判別モデルの学習における計算は、奈良県産業振興総合センターの「なら AI ラボ」の設備機器である Nvidia TESLA V100 (16GB) を 2 基を搭載したコンピュータを用いた。

2.6 検証

判別モデルの検証は、学習に用いていないテスト画像で実施する。学習用データと同様に、画像を読み込み、1440px×1920px に縮小した後、299px×299px の画像マスを切り抜き、学習した判別モデルを適用する。図 2 は、入力画像に判別モデルを適用した出力画像の例である。入力画像を縦 4 マス横 6 マスの格子状に区切り、各画像マスを判別モデルを適用した。この際、HSV 色空間で彩度が 30 に満たない画像は除外する。X をクラス名 (今回の場合は撮影日) とすると、判別モデルでは、入力されたある画像マス (mass) のクラスが X となる確率 $p_{mass}(X)$ が出力される。図

3 は、ある一つのマス A を拡大した画像である。この場合、赤字で示した「0726」(撮影日: 7 月 26 日) が 0.93 と最も確率が高いクラスとなり、 $p_A(X = "0726") = 0.93$ と表せる。ここで、N 枚の画像マスが存在する入力画像 M における全体確率 $P_{all}^M(X)$ は、次のとおりとする。

$$P_{all}^M(X) = \frac{\sum_{i=1}^N p_{mass_i}(X)}{N}$$

ここで、 $P_{all}^M(X)$ が最も大きくなるクラス X を出力画像の判別結果とする。

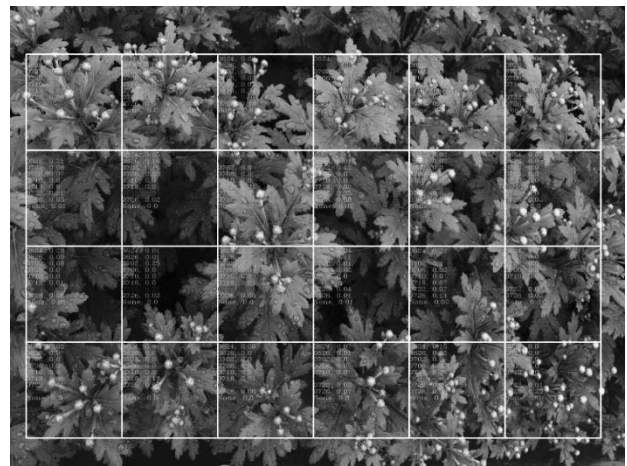


図 2 出力画像例

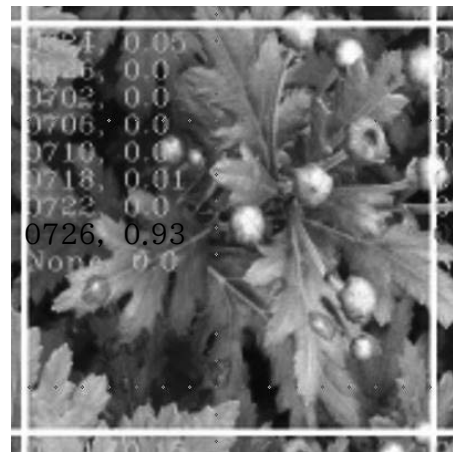


図 3 拡大した出力画像

3. 結果・考察

3.1 画像の収集と前処理

奈良県農業研究開発センターにおける試験圃場にて、小ギクの品種「春日の紅」「小鈴」の画像を収集した。表 2 及び表 3 にそれぞれの収集した画像、テスト画像、学習画像、前処理画像の各クラスのファイル数を示す。前処理画像は、学習画像に 2.2 の前処理を行った結果の枚数である。また、前処理後の画像を以下図 4 及び図 5 に示す。

表 2 「春日の紅」各クラスのファイル数

クラス名 (撮影日)	テスト用 画像	学習用 元画像	前処理後 画像
2019年6月24日	7	62	4286
2019年6月26日	7	39	2708
2019年7月2日	3	10	700
2019年7月6日	5	35	2450
2019年7月10日	5	31	2170
2019年7月18日	6	37	2590
2019年7月22日	6	35	2450
2019年7月26日	6	37	2587

表 3 「小鈴」各クラスのファイル数

クラス名 (撮影日)	テスト用 画像	学習用 元画像	前処理後 画像
2019年6月24日	5	46	3156
2019年6月26日	5	53	3684
2019年6月28日	5	38	2655
2019年6月30日	5	34	2375
2019年7月2日	5	35	2443
2019年7月4日	5	34	2380
2019年7月6日	5	41	2869
2019年7月8日	5	37	2590
2019年7月10日	5	30	2100
2019年7月12日	5	39	2730
2019年7月14日	5	40	2798
2019年7月16日	5	40	2800
2019年7月18日	5	33	2310

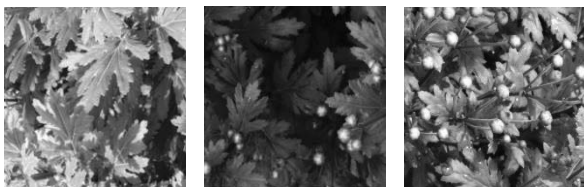


図 4 前処理後画像 (春日の紅)



図 5 前処理後画像 (小鈴)

3.2 学習

各クラスの学習データ数を統一するため、「春日の紅」では、学習データ数を700とし、8クラスの分類、2000回の繰り返し回数で、学習時間は6時間程度であった。また、「小鈴」では、学習データ数を2000とし、13クラスの分類、2000回の繰り返し回数で、学習時間は20時間程度であった。それぞれの学習曲線を図6に示す。いずれも正答率 accuracy が1へ近づいていき、2000epoch あたりでほぼ

頭打ちとなった。また、その際に loss が増えていないため、過学習が起こっていないことを確認できた。

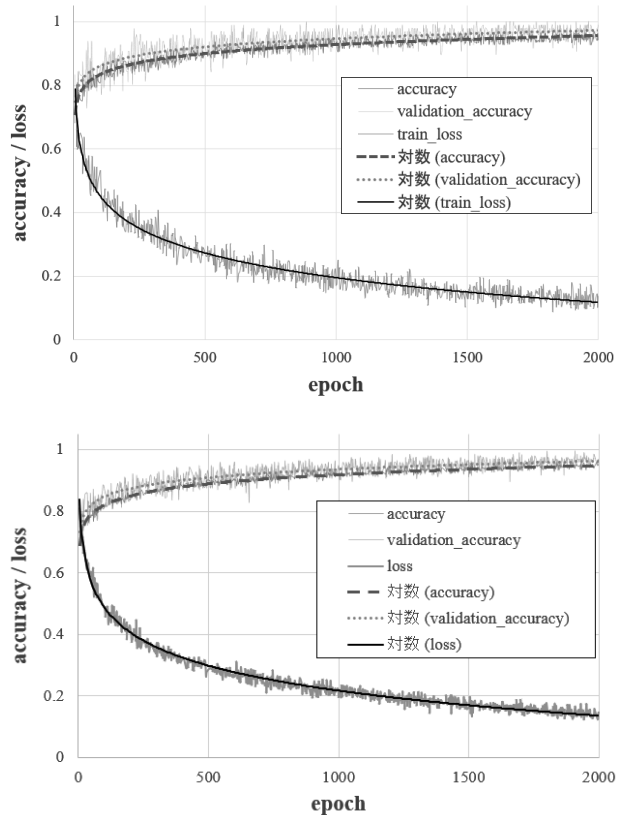


図 6 学習曲線 (上：春日の紅、下：小鈴)

3.3 検証

判別モデルの検証は、学習に用いていないテスト画像で実施した。2.6のとおり、テスト画像を画像マスに区切り、判別モデルを適用した。画像マスでの判別結果の混同行列を表4に示す、また、入力画像全体での判別結果の混同行列を表5に示す。混同行列の行は、判別モデルが判別したクラスで、列は真のクラスである。対角線上の濃色のセルは、正しく分類されたもので、対角線外のセルは、誤って分類されたものとなる。右端の列の値は、正しく分類された例の割合を示す。

春日の紅について、45枚のテスト画像を1075枚の画像マスに区切り、判別モデルを適用した。画像マスでの推定精度は1075枚中507枚が正解となり47.2%となった。また、入力画像全体での推定精度は45枚中27枚正解の60.0%となった。

小鈴について、65枚のテスト画像を1547枚の画像マスに区切り、判別モデルを適用した。画像マスでの推定精度は1547枚中582枚正解の37.6%となった。また、入力画像全体での推定精度は64枚中34枚正解の52.3%となった。

表 4 画像マスでの判別結果の混同行列
(上: 春日の紅、下: 小鈴)

		Prediction class								
		6/24	6/26	7/2	7/6	7/10	7/18	7/22	7/26	
True class	6/24	118	32	0	6	1	0	0	0	0.75
	6/26	33	125	0	8	1	0	0	0	0.75
	7/2	19	22	22	7	0	1	1	0	0.31
	7/6	10	19	0	86	1	4	0	0	0.72
	7/10	39	46	0	26	15	1	0	0	0.12
	7/18	10	37	0	10	7	78	0	2	0.54
	7/22	28	9	2	2	0	55	22	26	0.15
	7/26	46	11	0	3	0	43	0	41	0.28

		Prediction class														
		6/24	6/26	6/28	6/30	7/2	7/4	7/6	7/8	7/10	7/12	7/14	7/16	7/18		
True class	6/24	54	11	0	20	0	0	3	11	15	0	0	0	0	0.47	
	6/26	15	62	0	10	0	0	11	0	18	0	0	0	0	0.53	
	6/28	4	13	11	13	0	12	30	18	12	7	0	0	0	0.09	
	6/30	0	5	0	91	0	0	14	1	8	1	0	0	0	0.76	
	7/2	0	0	0	48	5	30	15	6	9	7	0	0	0	0.04	
	7/4	0	0	0	2	0	16	39	7	30	23	0	3	0	0.13	
	7/6	0	0	0	0	0	0	60	17	36	7	0	0	0	0.50	
	7/8	3	0	0	0	0	0	25	69	12	9	0	1	0	0.58	
	7/10	0	0	0	1	0	0	16	24	53	16	0	10	0	0.44	
	7/12	1	0	0	0	0	0	18	15	31	49	0	5	1	0.41	
	7/14	0	0	0	0	0	8	16	0	5	30	3	36	22	0.03	
	7/16	0	0	0	0	0	0	4	7	15	5	0	85	2	0.72	
	7/18	0	0	0	0	0	0	3	1	9	5	0	78	24	0.20	

表 5 入力画像全体での判別結果の混同行列
(上: 春日の紅、下: 小鈴)

		Prediction class								
		6/24	6/26	7/2	7/6	7/10	7/18	7/22		7/26
True class	6/24	7	0	0	0	0	0	0	0	1.0
	6/26	0	7	0	0	0	0	0	0	1.0
	7/2	1	1	1	0	0	0	0	0	0.3
	7/6	0	0	0	5	0	0	0	0	1.0
	7/10	0	4	0	1	0	0	0	0	0.0
	7/18	0	1	0	0	0	5	0	0	0.8
	7/22	1	0	0	0	0	5	0	0	0.0
7/26	3	0	0	0	0	1	0	2	0.3	

		Prediction class														
		6/24	6/26	6/28	6/30	7/2	7/4	7/6	7/8	7/10	7/12	7/14	7/16	7/18		
True class	6/24	3	0	0	2	0	0	0	0	0	0	0	0	0	0.6	
	6/26	1	4	0	0	0	0	0	0	0	0	0	0	0	0.8	
	6/28	0	0	1	0	0	3	1	0	0	0	0	0	0	0.0	
	6/30	0	0	0	5	0	0	0	0	0	0	0	0	0	1.0	
	7/2	0	0	0	4	0	1	0	0	0	0	0	0	0	0.0	
	7/4	0	0	0	0	0	4	0	1	0	0	0	0	0	0.0	
	7/6	0	0	0	0	0	0	4	0	1	0	0	0	0	0.8	
	7/8	0	0	0	0	0	0	0	4	0	1	0	0	0	0.8	
	7/10	0	0	0	0	0	0	0	1	4	0	0	0	0	0.8	
	7/12	0	0	0	0	0	0	0	0	1	4	0	0	0	0.8	
	7/14	0	0	0	0	0	0	0	0	0	3	0	1	1	0.0	
	7/16	0	0	0	0	0	0	0	0	0	0	0	5	0	1.0	
	7/18	0	0	0	0	0	0	0	0	0	0	0	4	1	0.2	

いずれの結果においても6割以下の正答率となった。正答率が6割以下となった大きな理由として考えられるのは、学習データのクラス分類に誤りが含まれていることが挙げられる。今回の検証では、多種多様な品種や環境条件で栽培を行う農業現場での作業性を考慮し、現場で撮影を行い登録するだけで学習用データを作成できるようにするため、同一撮影日の画像はすべて同一のクラスとした。しかし、実際には、同じ撮影日(クラス)において、生育状況が全く同じになるとは限らない。前処理後画像を確認すると、他の撮影日と区別がつかない画像が多数存在しており、学習用データを撮影日でのクラス分けだけではなく、画像の特徴量から分類するなどの別手法との組み合わせを検討する必要があると考えられる。

4. 結言

本研究では、ディープラーニングを活用するための前処理方法の検討及び、学習済みのモデルの一部を変更して再学習するファインチューニングと呼ばれる手法の適応を、農作物である小ギクの生育状況の画像判別において試みた。判別結果の正答率は6割程度となり、小菊の成長ステージ判断への適用には更なる検討が必要であることが分かった。引き続き、別手法の組み合わせなど、現場作業者の負担の少ないシステムを目指し改良を続ける。また、今回得た知見を活かすため、ディープラーニングによる画像認識の適用が、他の事例でも簡便に行えるように、システムの整備も進めていく。

参考文献

- 1) “Keras Documentation”, <https://keras.io/ja/>, (参照 2020-4-30)
- 2) “Keras backend”, <https://keras.io/ja/backend/>, (参照 2020-4-30)
- 3) “OpenCV”, <https://docs.opencv.org/master/index.html>, (参照 2020-4-30)
- 4) “画像分類モデルの使用例 Fine-tune InceptionV3 on a new set of classes”, https://keras.io/ja/applications/#_2, (参照 2020-4-30)
- 5) “InceptionV3”, <https://keras.io/ja/applications/#inceptionv3>, (参照 2020-4-30)
- 6) “ImageDataGenerator”, <https://keras.io/ja/preprocessing/image/>, (参照 2020-4-30)