

CNN による星図データに基づく星座識別手法の提案

1 はじめに

近年、機械学習の急速な発展を背景にして、人工知能 (Artificial Intelligence : AI) を用いた認識技術が注目を集めている。特に画像認識の分野においては盛んに研究成果が報告されており、ニューラルネットワークの登場によって広範なタスクにおいて高い精度での物体認識が可能となってきた。一方で、天体写真は対象の天体とそれ以外の領域の線引きが難しく、また撮影者や機材の違いによって写っている星の状態が大きく異なるため、天体写真から星座を認識するのは AI をもってしても非常に困難な課題となっている。以上の点を踏まえて本実験では、天体写真を人間のように認識する AI の構築を最終目的とする研究の初期段階として、天体写真に写っている星座を複数の候補の中から見つけ出す識別器の作成を目的とする。実験の具体的な手順としては、まず星座の訓練データを拡充するために独自のデータセットを作成し、作成したデータセットを複数の畳み込みニューラルネットワーク (Convolution Neural Network : CNN) [1] で学習させ、天体写真の識別を試みた。

2 要素技術

2.1 VGG

VGG [2] とは、2014 年に Oxford 大学の研究チームによって提案された CNN モデルの一つであり、13 層の畳み込み層と 3 層の全結合層の合計 16 層からなる VGG16 や、16 層の畳み込み層と 3 層の全結合層の合計 19 層からなる VGG19 などがある。2014 年の画像認識大会で好成績を収めたことからその後のモデルアーキテクチャに広く取り入れられている。また ImageNet と呼ばれる大規模画像データセットで学習済みのモデルが公開されており、このモデルを転移学習してほかの画像認識タスクに適用することの有効性が報告されている。

2.2 ResNet

ResNet [3] とは、2015 年に Microsoft Research にて提案された CNN モデルの一つであり、学習する層をスキップする Skip Connection の導入によ

り、層を深くするにつれて勾配が消失する問題を解決した。これにより 50 層や 152 層といった深さのネットワークを実現し、画像認識の精度を飛躍的に向上させた。VGG と同様に ImageNet で学習済みのモデルが公開されている。

3 実験

本実験で識別対象とした星座は、はくちょう座、ふたご座、いて座、カシオペア座、こぐま座、こと座、おおぐま座、おおいて座、オリオン座、おとめ座、おうし座、ペガサス座、さそり座、しし座、わし座の計 15 種類である。なおこれらの星座は筆者が独自に選定した。ここで、前述のほとんどの星座はモデルの学習に十分な枚数の画像を集めることができなかったため、モデルが各星座の形を認識するための独自のデータセットを作成し、モデルの訓練に活用することにした。なお本実験では深層学習ライブラリの keras と機械学習ライブラリの scikit-learn を用いて実験した。

3.1 独自データセットの作成

本実験で用いる独自データセットは、各星座を構成する星の位置、大きさおよび色の情報をもとに作成した。用いるデータとしては、欧州宇宙機関の位置天文衛星ヒッパルコスによる観測結果を元に編集され、公開されているヒッパルコス星表のデータを使用した。まず、アメリカ航空宇宙局 (NASA) のサイトより、ヒッパルコス星表に収録されている星の中で視等級¹が 7.0 未満の計 15404 個の星の赤経²、赤緯²、視等級、スペクトル型³の情報を獲得した。ここで、赤経と赤緯は赤道座標⁴であるため地上から見える星の並びを実現するためには地平座標に変換する必要がある。以下の (1) ~ (3) 式で赤経 α と赤緯 δ は地平座標へと変換される。ここで赤経 α と赤緯 δ はラジアン表記であるとする。また r は投影天球の半径であり、本実験ではこれを 1000 とした。なお x 軸は春分点の方向、 z 軸は天の北極の方向である。

¹肉眼または望遠鏡で見た天体の明るさから決めた等級

²天体の位置を表す値

³恒星が放つスペクトルによって分類されるもので色情報を含む

⁴天体の位置を表す天球座標系の一つ

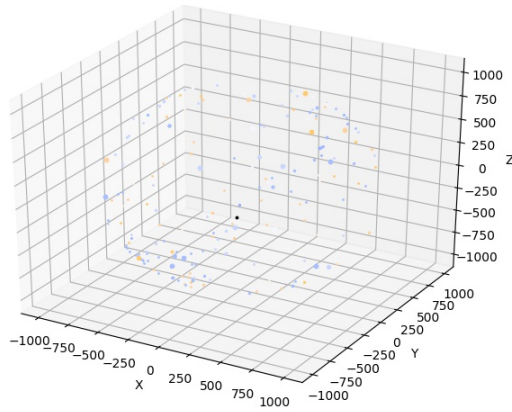


図 1: 三次元空間のプロット図

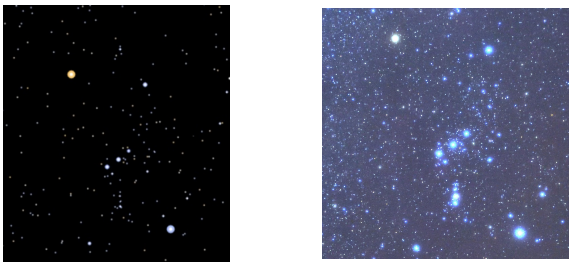


図 2: オリオン座の星図 図 3: オリオン座の写真

$$x = r \cos \alpha \cos \delta, \quad (1)$$

$$y = r \sin \alpha \cos \delta, \quad (2)$$

$$z = r \sin \delta. \quad (3)$$

これによって計算された地平座標を基に、三次元空間上に各星をプロットした。なお投影天球の半径 1000 に対し星の直径は基準を 15 / (視等級) とし、星の色はスペクトル型を反映したものとする。図 1 にプロット図を示す (スペースの都合上一部の星のみ表示)。この空間の一部を切り取ったものをデータとして扱う。具体的にはこの空間に対して、各星座のおおよその座標と原点との関係性からそれぞれ任意の平面を定め、そこに星を投影させる。なおこのとき、投影する平面の大きさにあわせて星の直径を微調整するものとする。またこの平面は各星座の大きさを考慮して筆者がそれぞれ独自で決定したものであり、平面自体はすべての画素値が 0 である黒色画像である。最後に、星を投影した平面に対してガウシアンフィルタをかけて完成とする。以降、作成したものを星図と呼ぶ。図 2、図 3 に完成したオリオ

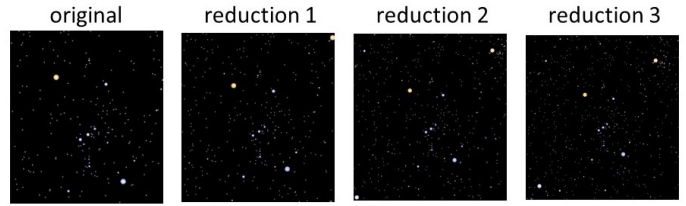


図 4: 異なる大きさの平面で作成された星図

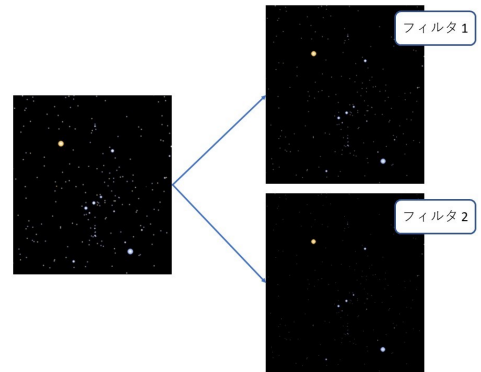


図 5: フィルタ処理

ン座の星図と、比較のためオリオン座の天体写真を示す。

この方法を用いて、切り取る平面の大きさを変えて各星座それぞれ 4 種類の星図を作成し、写っている星座の大きさが大きい写真から小さい写真まで幅広く対応できるようにした。図 4 に例としてオリオン座の場合を示す。

また、完成した星図に対して、

フィルタ 1: 縮小処理 (カーネルサイズ: 2×2)

フィルタ 2: 縮小処理 (カーネルサイズ: 3×3)

の二種類のフィルタをかけたものを追加で作成した。図 5 にオリオン座にフィルタ処理をした結果を示す。このフィルタ処理により、星の数が多写真から少ない写真まで幅広く対応できるようにした。

これらに加え、それぞれの星図について $-180^\circ \sim 150^\circ$ の範囲で 30° 間隔で回転させた回転画像計 12 枚を作成した。また天体写真の種類によっては色情報が少ないものもあることを考慮してそれぞれのグレースケール画像も作成した。以上より、各星座 (15 種類) につき 4 (切り取る平面の大きさの違い) \times 3 (フィルタ処理) \times 12 (回転画像) \times 2 (RGB スケールおよびグレースケール) の 288 枚、合計 4320 枚の星図を作成した。

表 1: 実験パラメータ

最適化関数	SGD
学習率	1.0×10^{-4}
損失関数	binary cross entropy
エポック数	100
バッチサイズ	16

表 2: 識別結果

		予測値	
		星図	天体写真
真 値	星図	42	3
	天体写真	3	42

3.2 予備実験

作成した星図を用いて識別実験をする前に、星図と天体写真にどれほどの差があるのかを確認するため、星図と天体写真の2クラスを識別する予備実験をした。星図の訓練画像には各星座につき図4における original 画像にフィルタ処理と回転処理をした 3×12 枚の36枚、合計540枚を、天体写真の訓練画像には各星座3枚ずつ、合計45枚をスクレイピングによって収集し、それらに星図作成時に行った回転処理をした合計540枚を使用した。また星図のテスト画像には各星座につき図4における reduction 3 画像にフィルタ処理をした合計45枚を、天体写真のテスト画像には各星座3枚ずつ、合計45枚を訓練画像とは別にスクレイピングによって収集したものを使用した。なお天体写真に関しては撮影者の編集の都合により白く霞がかっているものもあり、星図との類似性が小さくなることを考慮して、画像の輝度に関して閾値を設け、それよりも小さな輝度をもつピクセルは黒塗りする処理を施した。

本実験では、VGG16 を用いて識別を試みた。初期重みとしては ImageNet で学習されたモデルを使用し、15 層目以降をファインチューニングした。なお画像サイズは 128×128 とした。表1に実験パラメータを示す。また表2に識別結果の混同行列を示す。これより、星図と天体写真には明確な差があることがわかる。この差を小さくするために、keras ライブラリに用意された Data Augmentation 用のクラスである ImageDataGenerator を天体写真の識別用に改良した独自の ImageDataGenerator を作成することを試みた。



図 6: 独自の ImageDataGenerator の例

表 3: 識別結果 (ImageDataGenerator あり)

		予測値	
		星図	天体写真
真 値	星図	43	2
	天体写真	9	36

3.3 独自の ImageDataGenerator の作成

keras の ImageDataGenerator には、回転画像や切り抜き画像を生成する関数などがデフォルトで用意されている。本実験ではそこに一定の確率でノイズをかける関数と、一定の確率でマスクを付与する関数を追加した。なおマスクに関しては場所や大きさ、画素値が適用ごとにランダムで設定されるようにした。これは天体写真に写りこむ天の川や山影などに対応することを期待している。本実験ではノイズをかける確率を 0.5、マスクを付与する確率を 0.2 とした。図6に独自の ImageDataGenerator をかけた際の例を示す。

表3に独自の ImageDataGenerator を使って予備実験と同じ実験をした時の結果の混同行列を示す。星図と識別された天体写真の数が増えており、これにより、独自の ImageDataGenerator を使うことでわずかではあるが星図と天体写真の差が縮まったと考えられる。

3.4 複数の CNN を用いた多クラス識別実験

3.1 で作成した星図と 3.3 で作成した ImageDataGenerator を用いて複数の CNN で天体写真の多クラス識別を試みる。本実験では CNN モデルとして VGG16, VGG19, ResNet50 を用いた。全モデルにおいて ImageNet で学習された初期重みを用いており、VGG16 では 15 層目以降を、VGG19 では 17 層目以降をファインチューニングした。

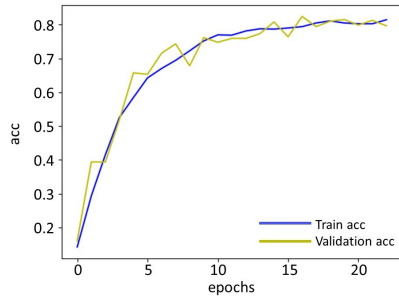


図 7: VGG19 を用いた際の accuracy の推移

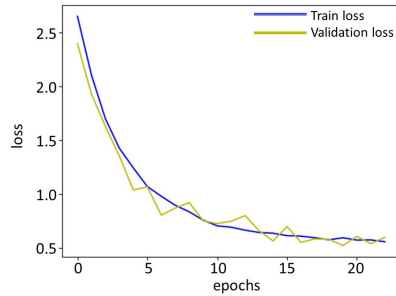


図 8: VGG19 を用いた際の loss の推移

訓練画像には作成した全星図 4320 枚を, テスト画像には予備実験でテスト画像として使用した天体写真 45 枚を使用した. 実験パラメータおよび画像サイズ, 天体写真に施す閾値処理は予備実験と同じとした. 図 7, 図 8 に例として VGG19 の訓練および検証の際の accuracy と loss の推移を示す. なお横軸は epoch 数, 縦軸は accuracy または loss の値を表している. VGG19 を用いた際は過学習を防ぐために設定した Early stopping のために, 23 epoch 目で学習が停止した.

訓練時と検証時の accuracy がどちらも徐々に増加しており, また loss の値がどちらも徐々に減少していることから学習は進んでいると考えられる. 表 4 に各 CNN での Top 1 accuracy および Top 3 accuracy を示す. なお比較のために同じ実験条件で自作の ImageDataGenerator を使わなかった場合の実験結果も示している. ベースラインは 15 クラス分類であることを考慮して Top 1 accuracy では 1/15, Top 3 accuracy では 1/5 で設定している. また accuracy の各値は小数第四位を四捨五入している.

表 4 より, 実験に用いたすべての CNN による識別で accuracy がベースラインを超えていることから, 独自の星図データセットを用いた星座識別手法の有効性を確認することができた. また, 独自の Image-

表 4: 各 CNN における識別結果

	Top 1 accuracy	Top 3 accuracy
VGG16 (ImageDataGenerator あり)	0.267	0.489
VGG16	0.222	0.422
VGG19 (ImageDataGenerator あり)	0.333	0.556
VGG19	0.267	0.467
ResNet50 (ImageDataGenerator あり)	0.222	0.467
ResNet50	0.133	0.311
ベースライン	0.067	0.200

DataGenerator を用いた方が全体的に accuracy が高くなっていることから星図と天体写真の差を縮めることができていると考えられ, 独自の ImageDataGenerator の有効性も確認することができた.

4 まとめと今後の課題

本実験により, 独自の星図データセットおよび独自の ImageDataGenerator を用いた CNN による天体写真の識別手法の有効性を確認することができた. また本実験により星図を天体写真に近づけることで accuracy が上昇することが確認されたため, 今後はより星図を天体写真に近づける手法について検討していきたい.

参考文献

- [1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, #jun# 2017.
- [2] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015.
- [3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.