

# Report

## 利用した実データの説明:

この画像セットはウェブサイトからダウンロードした食物画像です。総計 11 個種類の食物があります。この画像セットは三つの部分があります。Training data には 9866 枚、Validation data には 3430 枚、Testing data には 3347 枚の画像があります。

## タスク:

Training data と Validation data を利用して、Testing data を分類すること。

## 実現した知的機械の概要:

画像分類は CNN を利用しましたが、精度を上げるために、DCGAN を使って画像 (200 枚 \* 11 種類) を生成しました。

## 考えられる応用先:

主には食物、植物などの分類と認識の分野で利用できます。ドローンを使って空撮した農地画像の分析を研究していますので、知能情報の技術を用いて、画像から農地情報をモニタリングするように頑張っています。今の時代で、農業 IOT を利用して、農地情報を収集していますが、ほとんどは数字データです。画像データが少なく、機械学習モデルに入力した時、データ量が少ないため、学習精度がなかなか上がりませんでした。さまざまな GAN モデルを利用して、画像を生成すれば、学習精度を向上になりました。

## 利用した手法の説明:

1. まず、python パッケージとデータセットを読み込みます。

```
[ ] import os
import numpy as np
import cv2
import torch
import torch.nn as nn
import torchvision.transforms as transforms
import pandas as pd
from torch.utils.data import DataLoader, Dataset
import time

[ ] workspace_dir = '.'
!gdown --id '19CzXudqN58R3D-1G8KeFWk8UDQw1b8is' --output "{workspace_dir}/food-11.zip"
!unzip -q "{workspace_dir}/food-11.zip" -d "{workspace_dir}/"

[ ] Downloading...
From: https://drive.google.com/uc?id=19CzXudqN58R3D-1G8KeFWk8UDQw1b8is
To: /content/food-11.zip
1.16GB [00:09, 123MB/s]
```

2. データセットを定義します。画像を保存するために、torchvision を使用します。ここ

では cv2 で読み取った画像 (BGR) を torchvision フォーマット (RGB) に変換します。

```
[ ] from torch.utils.data import Dataset, DataLoader
import cv2
import os

class FoodDataset(Dataset):
    def __init__(self, fnames, transform):
        self.transform = transform
        self.fnames = fnames
        self.num_samples = len(self.fnames)
    def __getitem__(self, idx):
        fname = self.fnames[idx]
        img = cv2.imread(fname)
        img = self.BGR2RGB(img) #because "torchvision.utils.save_image" use RGB
        img = self.transform(img)
        return img

    def __len__(self):
        return self.num_samples

    def BGR2RGB(self, img):
        return cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
```

3. 画像生成する前に、入力画像のサイズを (64, 64) に変更し、標準化に計算します。。

- ①. resize the image to (64, 64)
- ②. linearly map [0, 1] to [-1, 1]

```
import glob
import torchvision.transforms as transforms

def get_dataset(root):
    fnames = glob.glob(os.path.join(root, '*'))
    transform = transforms.Compose(
        [transforms.ToPILImage(),
         transforms.Resize((64, 64)),
         transforms.ToTensor(),
         transforms.Normalize(mean=[0.5] * 3, std=[0.5] * 3) ] )
    dataset = FoodDataset(fnames, transform)
    return dataset
```

4. DCGAN を base mode として、モデルを構造します。(詳しい code はここに表示していません、ソースコードファイルにチェックしてください。)
5. 各タイプの画像をトレーニングして、モデルを生成します。

パラメータは以下のように設定します。

batch\_size = 8, z\_dim = 100, lr = 1e-4, n\_epoch = 50

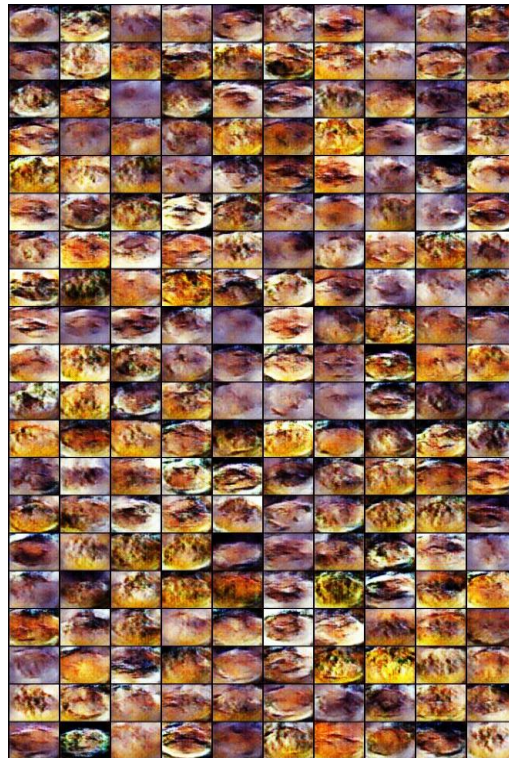
loss criterion: criterion = nn.BCELoss()

optimizer:

opt\_D = torch.optim.Adam(D.parameters(), lr=lr, betas=(0.5, 0.999))

opt\_G = torch.optim.Adam(G.parameters(), lr=lr, betas=(0.5, 0.999))

6. 結果を出す。('0'という種類の結果は以下のように出力します。)



7. 生成した画像をダウンロードして、トレーニングデータセットに追加します。CNNを利用して、モデルを構造します。精度は90パーセント以上に上がる。

```
[023/030] 39.74 sec(s) Train Acc: 0.879128 Loss: 0.003332
[024/030] 39.98 sec(s) Train Acc: 0.859582 Loss: 0.003125
[025/030] 39.96 sec(s) Train Acc: 0.871540 Loss: 0.002829
[026/030] 39.92 sec(s) Train Acc: 0.884928 Loss: 0.002607
[027/030] 39.97 sec(s) Train Acc: 0.885153 Loss: 0.002468
[028/030] 39.97 sec(s) Train Acc: 0.897037 Loss: 0.002316
[029/030] 39.92 sec(s) Train Acc: 0.902903 Loss: 0.002172
[030/030] 40.12 sec(s) Train Acc: 0.920126 Loss: 0.001785
```

8. 最後に、学習したモデルを用いて、テストデータセットの画像の種類を判断します。".csv"ファイルを出力します。

	A	B	C
1	Id	Category	
2	0	3	
3	1	10	
4	2	0	
5	3	9	
6	4	7	
7	5	3	
8	6	9	
9	7	10	
10	8	3	
11	9	2	
12	10	0	
13	11	0	
14	12	0	
15	13	10	

### 工夫した点：

- (1) 画像生成する時、最初は全部の種類 of 食物データを一緒に用いて、画像を生成した  
いと思いますが生成した画像はすごくいやになってしまいました。だから、各種類  
の画像セットをそれぞれにトレーニングして、より高品質な画像を生成します。
- (2) GAN モデルを構造するとき、パラメータの選択は結構難しいと思います。
- (3) 各種類の画像セットをそれぞれにトレーニングする場合は、学習できる画像は少な  
くなって、生成した画像の解像度は低いです。自分学部生の時、開発した画像処理  
ソフトウェアを利用して、生成した画像を鮮鋭化に処理しました。
- (4) CNN を用いた時、一番いいパラメータの調整も時間かかるとおもいます。

### 最後の評価：

- (1) 生成モデルは結構難しいので、いまあまり上手ではないので、生成した画像の品質  
は高くないです。今後もっと頑張った方がいいと思います。
- (2) 今回 CNN の分類精度が 92% に達しました。結構いい結果だと思います。
- (3) 今後、自分の研究分野（農業）に知能情報技術を用いて、特定のタスクを解決する  
ことを目指します。

原田先生：ご指導ありがとうございました