Deep Learning

Blood Cell Classification

U Kang
Seoul National University

In This Lecture

- Blood cell image
- Multi-class classification from blood cell image
- Implement a deep learning model

Outline

- **Introduction**
- Dataset
- Preprocessing Codes

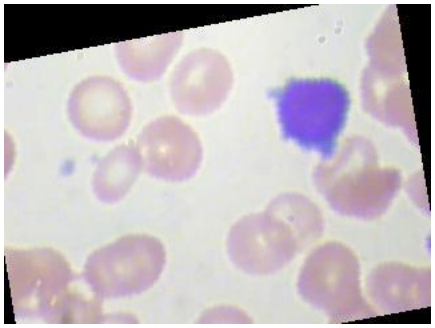
Motivation

- The diagnosis of blood-based diseases often involves identifying and characterizing patient blood samples
- Automated methods to detect and classify blood cell subtypes have important medical applications
  - Goal: detect blood cell type from a blood cell image

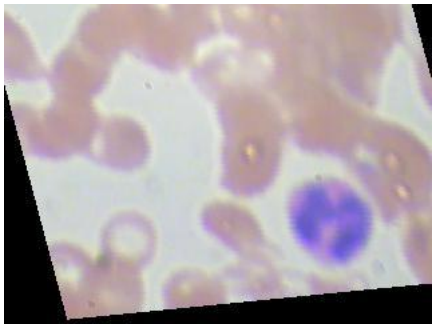
Goals

- Classify the blood cell image data into one of four categories

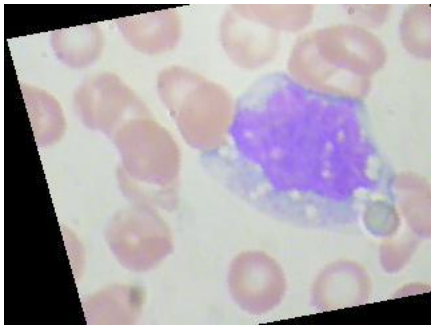
LYMPHOCYTE



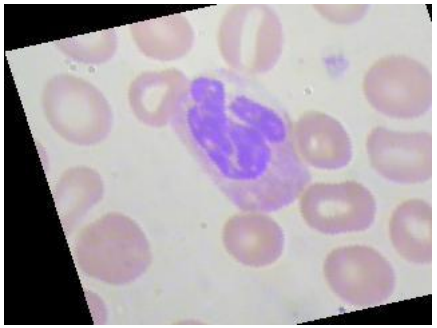
EOSINOPHIL



MONOCYTE



NEUTROPHIL



Problem Definition

- **Given:** blood cell image data

- **Classify:** the data into the correct categories
  - LYMPHOCYTE
  - EOSINOPHIL
  - MONOCYTE
  - NEUTROPHIL

Outline

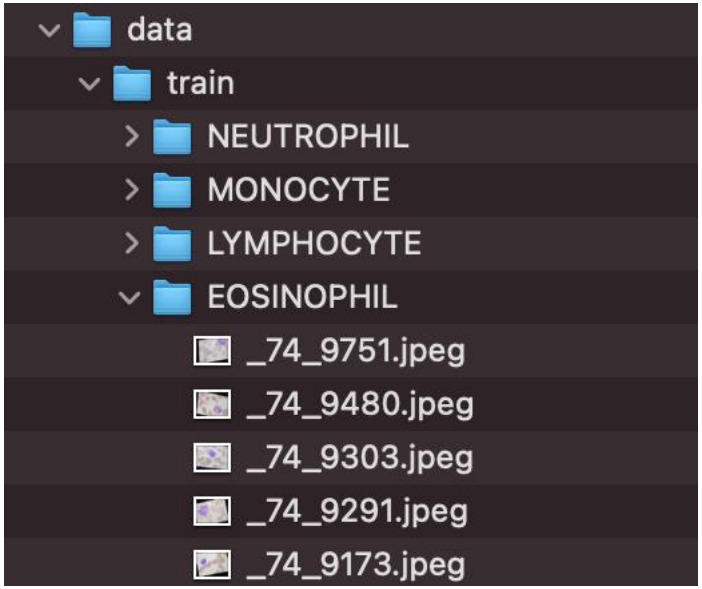
- ☑ Introduction
- ☐ **Dataset**
- ☐ Preprocessing Codes

Training Dataset

- Image size is (240, 320)
- Number of images in each category
  - LYMPHOCYTE: 2,483
  - EOSINOPHIL: 2,497
  - MONOCYTE: 2,478
  - NEUTROPHIL: 75

Training Dataset

- Approximately 7,500 images grouped into 4 different folders according to cell type
- JPEG files in training directory



Test Dataset

- 2,467 images
- The ratio of classes is balanced

Outline

- ☑ Introduction
- ☑ Data
- ☐ **Preprocessing Codes**

Import libraries

- Import the libraries such as numpy, pandas, math, cv2, os, matplotlib, tensorflow, and keras

```
import numpy as np
import pandas as pd
import math, cv2, os
from tensorflow.keras.preprocessing.image import ImageDataGenerator
import keras
%matplotlib inline
import matplotlib.image  as mpimg
import matplotlib.pyplot as plt
import tensorflow as tf

print(tf.__version__)
```

Loading the Dataset

- Using the 'ImageDataGenerator' library, we prepare the input data

```
train_dataset_path = "./data/train"

IMG_SIZE      = 128
BATCH_SIZE    = 1
CATEGORIES    = ['EOSINOPHIL','LYMPHOCYTE','MONOCYTE','NEUTROPHIL']

train_datagen = ImageDataGenerator()

train_generator = train_datagen.flow_from_directory(
        train_dataset_path,
        target_size=(IMG_SIZE, IMG_SIZE),
        color_mode='rgb',
        batch_size=BATCH_SIZE,
        shuffle=True,
        seed=4,
        class_mode="categorical")
```

```
Found 7533 images belonging to 4 classes.
```

Test

- Test measure
  - $acc = \frac{number\ of\ correct\ predictions}{total\ number\ of\ predictions}$
- Save results to csv file
  - 1st column: file names
  - 2nd column: predicted classes
    - Separator: comma ‘,’
    - Header: [‘Filename’, ‘Prediction’]

```
1 Filename,Prediction
2 _11_6757.jpeg,EOSINOPHIL
```

Submission

- Submit the “team#.zip” that contains followings:
  - “team#.csv”: result including prediction labels
  - “team#.ppt”: ppt summarizes your work
    - 10 min for presentation and 5 min for Q&A
    - Preprocessing, model architecture, training technique etc.
  - “team#.ipynb”: your code
- E-mail address: ant6si@snu.ac.kr
- Leaderboard [link]
- Submission Due: 2022/02/25 12:30

Questions?