CNN as Malaria Parasite Detection(D-detectron) by Sahil Khanna

Introduction

Malaria is a blood disease caused by the Plasmodium parasites transmitted through the bite of the female Anopheles mosquito. Microscopists commonly examine thick and thin blood smears to diagnose disease and compute parasitemia. However, their accuracy depends on smear quality and expertise in classifying and counting parasitized and uninfected cells.

Convolutional Neural Networks (CNN), a class of deep learning (DL) models promise highly scalable and superior results with end-to-end feature extraction and classification. Automated malaria screening using DL techniques could, therefore, serve as an effective diagnostic aid.

In my project, I'm going to build a ***CNN(D-detectron)*** toward classifying parasitized and uninfected cells to aid in improved disease screening cells and will evaluate the performance of D-detectron vs pre-trained CNN-based DL models.

Dataset description dataset source: NLM

The dataset is from the National Library of Medicine (NLM) hosts a repository of segmented cells from the thin blood smear slide images from the Malaria Screener research activity

The dataset contains a total of **27,558 cell images** with equal instances of parasitized and uninfected cells.

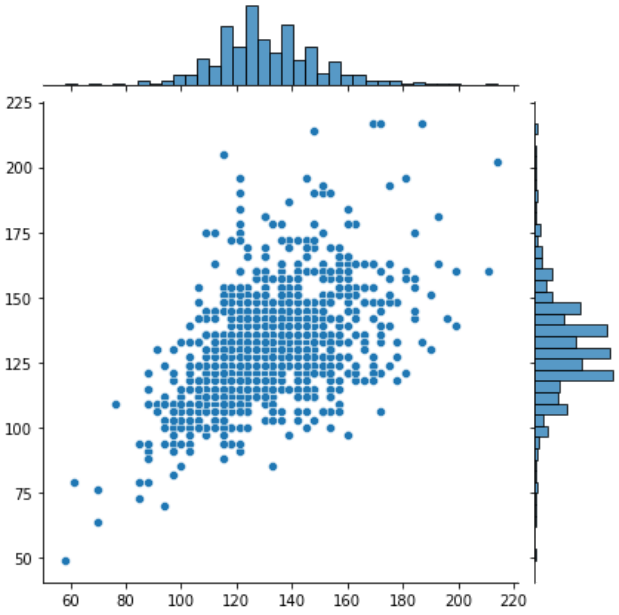
Objective

Using a deep neural network build a model to detect malaria parasites in thin-blood smear images

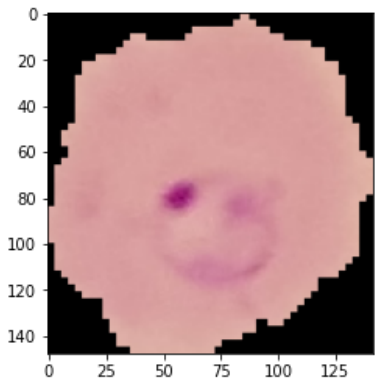
Preprocessing & Exploratory Data Analysis

Since images were acquired using a smartphone's built-in camera these images are of different sizes. I reshaped all the images with an average size of 130.

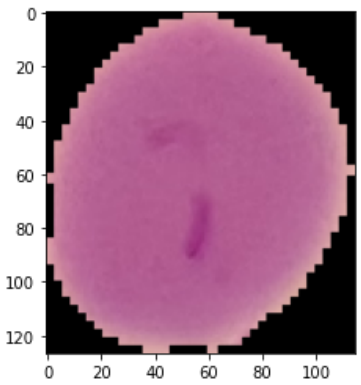
Visualization to capture average size



Parasitized Cell



Uninfected cell

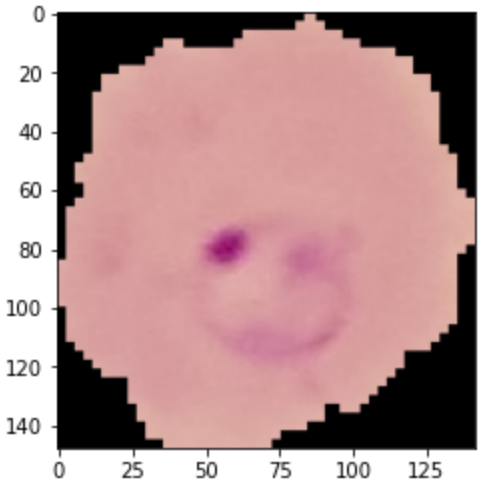


Preparing the Data for model

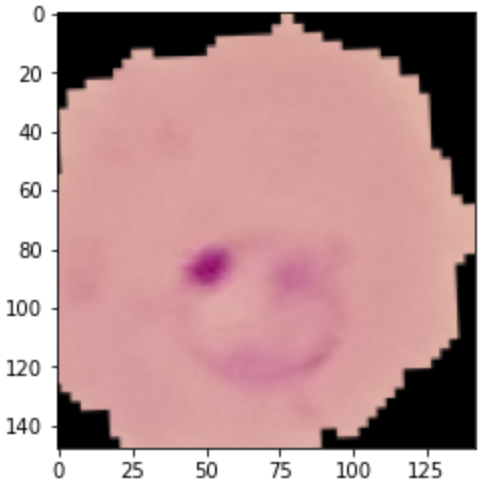
There is too much data for us to read all at once in memory. I used some built-in functions in Keras(Image data generator) to automatically process the data, generate a flow of batches (in the **size of 100 images**) from a directory, and also manipulate the images.

Image Manipulation; It's usually a good idea to manipulate the images with rotation, resizing, and scaling so the model becomes more robust to different images that our data set doesn't have. We can use the *ImageDataGenerator* to do this automatically for us.

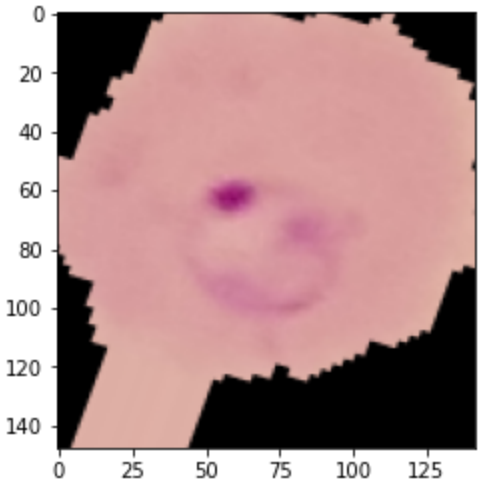
Original Image



Random rotation



Random shift



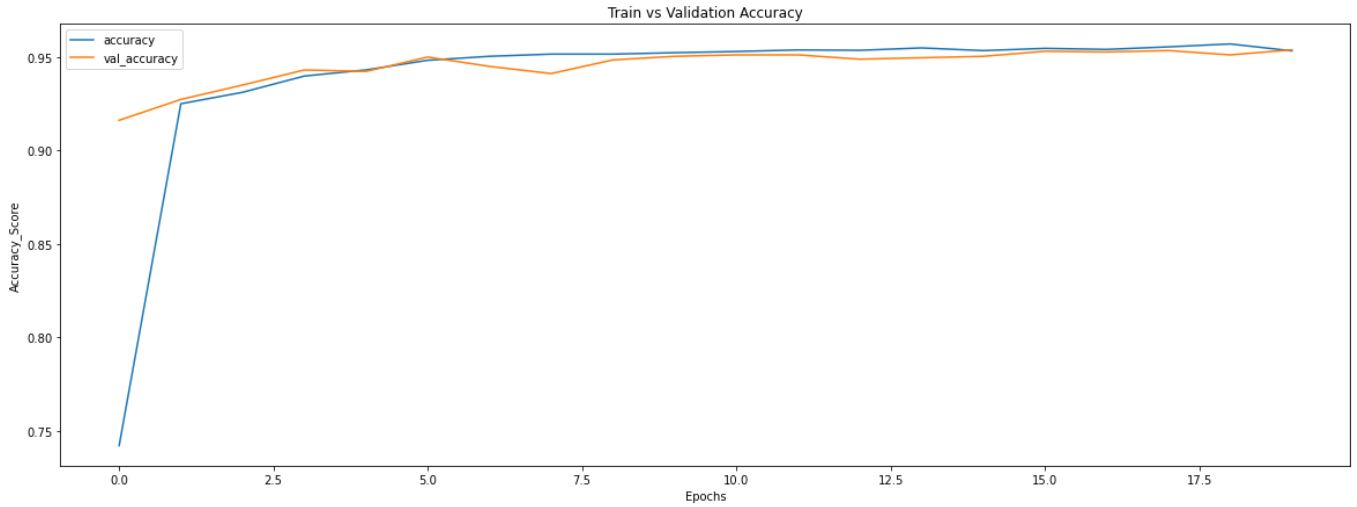
Model Architecture

The D-detectron has **3 convolution layers (filters 32, 64, 64)**. Each layer has a **MaxPool** layer of **(2,2)**, Kernel_size of 3x3, and **Activation function 'relu'**. One FC with **128 neurons** and 'relu' as the activation function, **Dropout layer (0.5)** and **Adam** optimizer. Output layer with **1 neuron** and **sigmoid** activation function.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 128, 128, 32)      896
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 64, 64, 32)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 62, 62, 64)        18496
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 31, 31, 64)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 29, 29, 64)        36928
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 14, 14, 64)        0
_________________________________________________________________
flatten (Flatten)            (None, 12544)             0
_________________________________________________________________
dense (Dense)                (None, 128)               1605760
_________________________________________________________________
activation (Activation)      (None, 128)               0
_________________________________________________________________
dropout (Dropout)            (None, 128)               0
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 129
_________________________________________________________________
activation_1 (Activation)    (None, 1)                 0
=================================================================
Total params: 1,662,209
Trainable params: 1,662,209
Non-trainable params: 0
_________________________________________________________________
```

Model Assessment

Train vs Test Accuracy



Classification Report

	Precision	Recall	F-Measure	class
	0.96	0.94	0.95	0
	0.94	0.97	0.95	1
Weighted Avg	0.95	0.95	0.95	

Confusion matrix

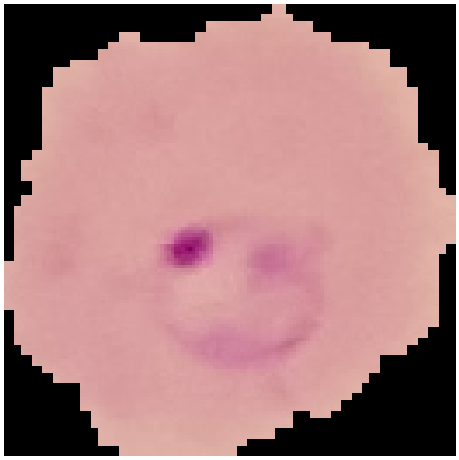
0	1	class
1223	77	0
45	1255	1

D-detectron vs Pre-trained

Models	Precision	Recall	F-Measure	Accuracy
AlexNet	0.941	0.947	0.944	0.944
VGG-16	0.969	0.949	0.959	0.959
ResNet-50	0.972	0.947	0.959	0.959
D-detectron	**0.952**	**0.955**	**0.954**	**0.954**

Prediction on new image

Test image



Model Prediction vs actual

```
In [53]: cnn.predict(my_image)
Out[53]: array([[0.]], dtype=float32)
```

```
In [51]: train_image_gen.class_indices
Out[51]: {'parasitized': 0, 'uninfected': 1}
```

Summary

Conclusion

The customized model converged to an optimal solution due to hyper-parameter optimization, implicit regularization, and data augmentation method imposed by smaller convolutional filter sizes and aggressive dropouts in the fully connected layers.

Future scope

I can use pre-trained convolutional neural networks as feature extractors toward improved Malaria parasite detection in thin blood smear images. I can also apply ensemble learning to reduce the model variance by optimally combining the predictions of multiple models.

Further, I want to explore D-detectron in the detection of other parasitic cell diseases like cancer.

References

[1]Performance evaluation of deep neural ensembles toward malaria parasite detection in thin blood smear images

Sivaramakrishnan Rajaraman, Stefan Jaeger, Sameer K. Antani

[2] Pre-trained convolutional neural networks as feature extractors toward improved malaria parasite detection in thin blood smear images

Sivaramakrishnan Rajaraman1, Sameer K. Antani1, Mahdieh Poostchi1, Kamolrat Silamut2, Md. A. Hossain3, Richard J. Maude2,4,5, Stefan Jaeger1, George R. Thoma1