Introduction

This report will analyze classification of 2 insect species, bees and butterflies, by building convolutional neural network model. The result of this report should provide foundation for refining more accurate image recognition capabilities on the given species well as potential to further expand the model on larger classes of animals.

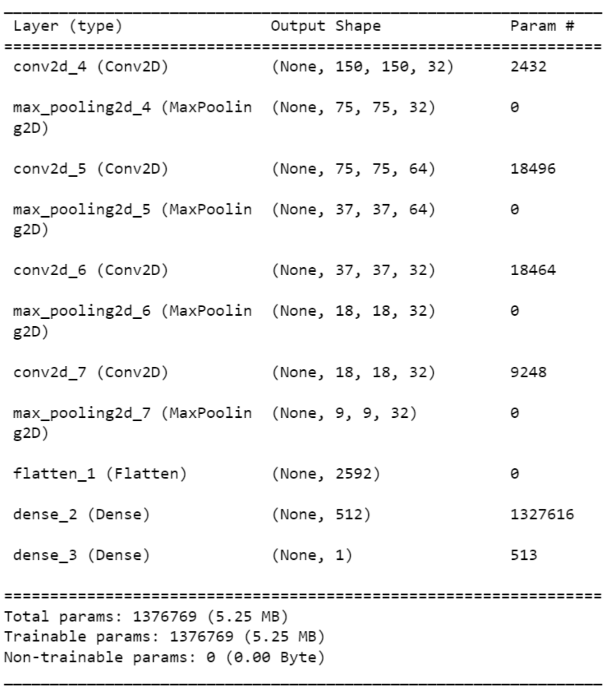
Preprocessing

The image data was obtained from Kaggle animal image dataset, which consist of 90 different species (https://www.kaggle.com/datasets/iamsouravbanerjee/animal-image-dataset-90-different-animals/).

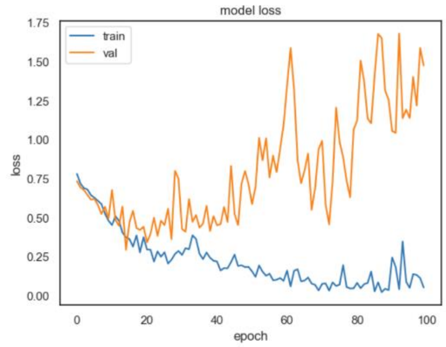
For the purpose of this analysis, only bees and butterflies images were used. Each set of species consist of 60 uncropped RGB images. The images were standardized to 150x150px using 1/255 rescaling factor 0.2 validation split. The samples were then split 8:2 for training and test.

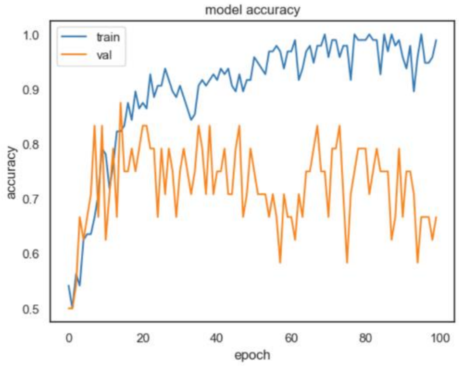
Model

The first models were built using 4 convolution layers followed by 1 dense layer with 512 neurons using relu as activation function for hidden layers and sigmoid for the output layer with adam optimizer. The model was then processed for 100 epochs with batch size of 20.



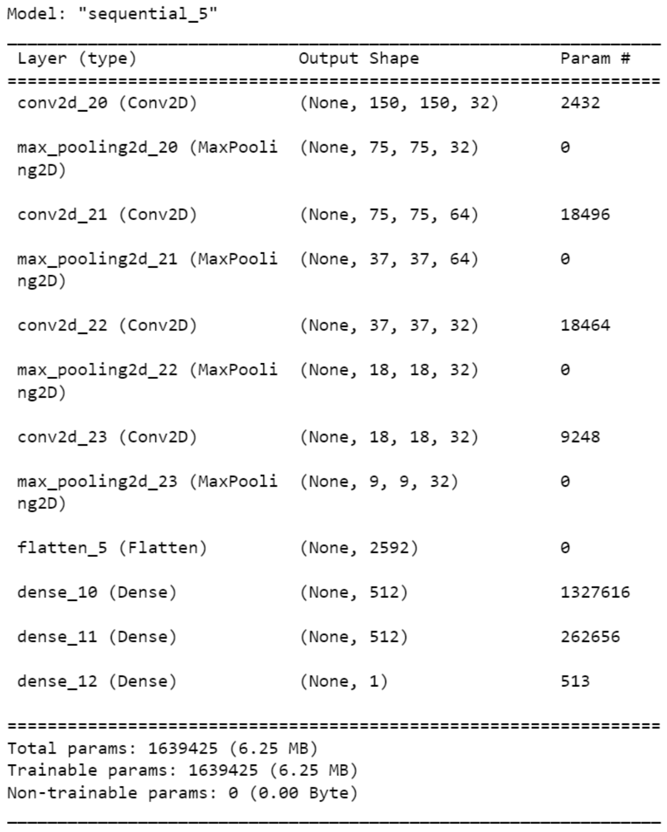
```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_4 (Conv2D)           (None, 150, 150, 32)      2432

 max_pooling2d_4 (MaxPoolin  (None, 75, 75, 32)        0
 g2D)

 conv2d_5 (Conv2D)           (None, 75, 75, 64)        18496

 max_pooling2d_5 (MaxPoolin  (None, 37, 37, 64)        0
 g2D)

 conv2d_6 (Conv2D)           (None, 37, 37, 32)        18464

 max_pooling2d_6 (MaxPoolin  (None, 18, 18, 32)        0
 g2D)

 conv2d_7 (Conv2D)           (None, 18, 18, 32)        9248

 max_pooling2d_7 (MaxPoolin  (None, 9, 9, 32)          0
 g2D)

 flatten_1 (Flatten)         (None, 2592)              0

 dense_2 (Dense)             (None, 512)               1327616

 dense_3 (Dense)             (None, 1)                 513

=================================================================
Total params: 1376769 (5.25 MB)
Trainable params: 1376769 (5.25 MB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```



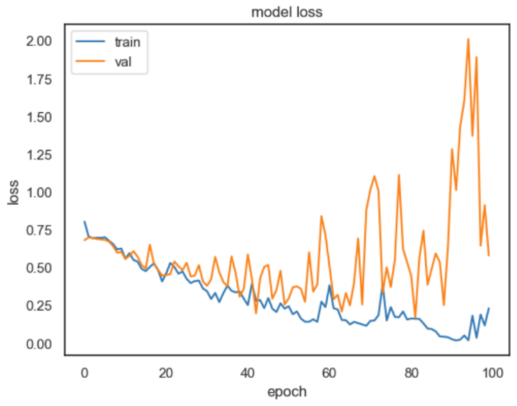


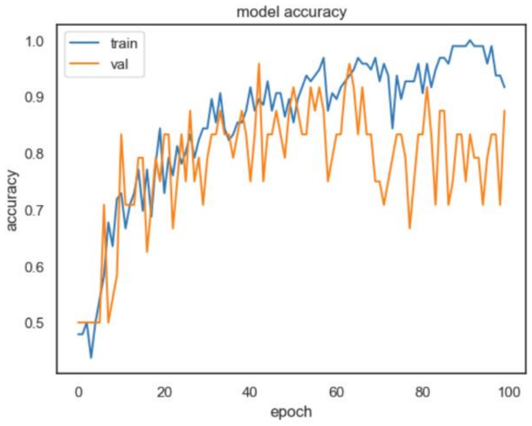
This model resulted in the accuracy score of 0.79167

The second model was built using the same convolution layers with 2 dense layers having 512 neurons each. LeakyReLU was used instead to account for dying ReLU problem. The model was trained for 100 epochs with the batch size of 1.



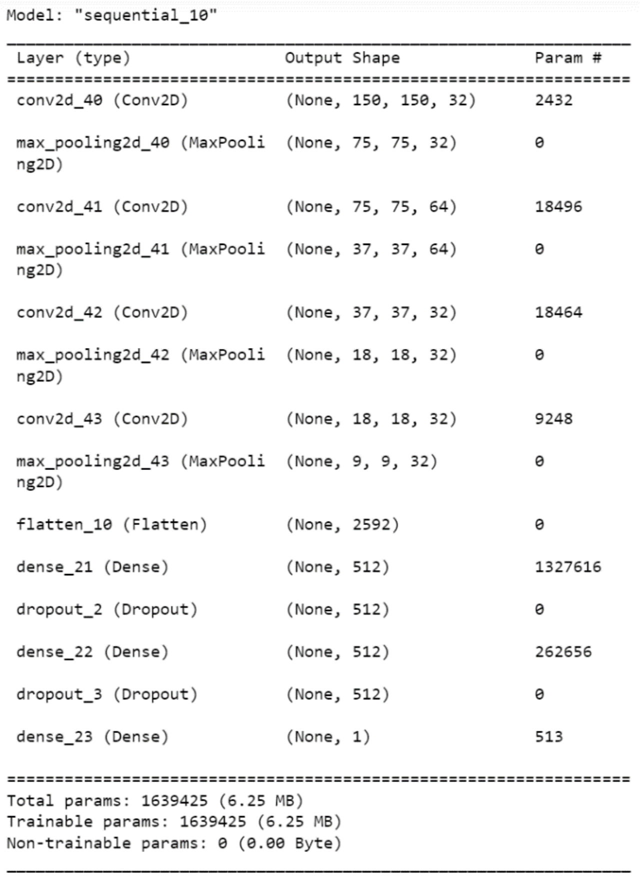
```
Model: "sequential_5"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 conv2d_20 (Conv2D)          (None, 150, 150, 32)      2432      
                                                                 
 max_pooling2d_20 (MaxPooli  (None, 75, 75, 32)        0         
 ng2D)                                                           
                                                                 
 conv2d_21 (Conv2D)          (None, 75, 75, 64)        18496     
                                                                 
 max_pooling2d_21 (MaxPooli  (None, 37, 37, 64)        0         
 ng2D)                                                           
                                                                 
 conv2d_22 (Conv2D)          (None, 37, 37, 32)        18464     
                                                                 
 max_pooling2d_22 (MaxPooli  (None, 18, 18, 32)        0         
 ng2D)                                                           
                                                                 
 conv2d_23 (Conv2D)          (None, 18, 18, 32)        9248      
                                                                 
 max_pooling2d_23 (MaxPooli  (None, 9, 9, 32)          0         
 ng2D)                                                           
                                                                 
 flatten_5 (Flatten)         (None, 2592)              0         
                                                                 
 dense_10 (Dense)            (None, 512)               1327616   
                                                                 
 dense_11 (Dense)            (None, 512)               262656    
                                                                 
 dense_12 (Dense)            (None, 1)                 513       
                                                                 
=================================================================
Total params: 1639425 (6.25 MB)
Trainable params: 1639425 (6.25 MB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```



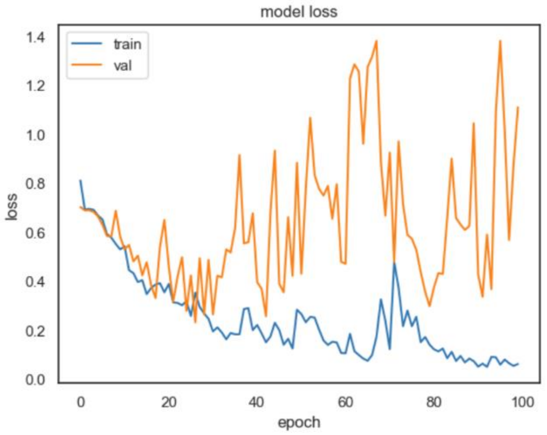


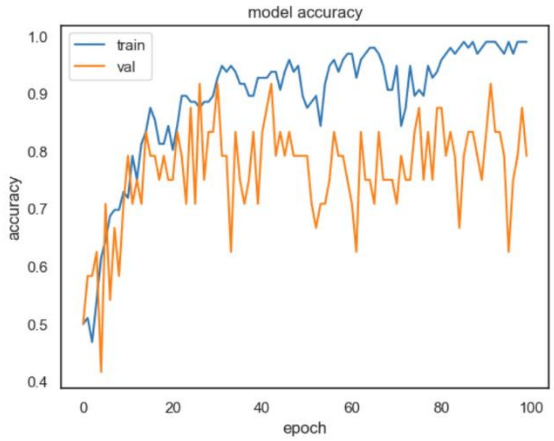
This model resulted in the accuracy score of 0.9583

The third model was built off of model 2 with regularization by dropout (=0.1). This model was trained for 100 epochs with batch size of 1



```
Model: "sequential_10"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_40 (Conv2D)          (None, 150, 150, 32)      2432

 max_pooling2d_40 (MaxPooli  (None, 75, 75, 32)        0
 ng2D)

 conv2d_41 (Conv2D)          (None, 75, 75, 64)        18496

 max_pooling2d_41 (MaxPooli  (None, 37, 37, 64)        0
 ng2D)

 conv2d_42 (Conv2D)          (None, 37, 37, 32)        18464

 max_pooling2d_42 (MaxPooli  (None, 18, 18, 32)        0
 ng2D)

 conv2d_43 (Conv2D)          (None, 18, 18, 32)        9248

 max_pooling2d_43 (MaxPooli  (None, 9, 9, 32)          0
 ng2D)

 flatten_10 (Flatten)        (None, 2592)              0

 dense_21 (Dense)            (None, 512)               1327616

 dropout_2 (Dropout)         (None, 512)               0

 dense_22 (Dense)            (None, 512)               262656

 dropout_3 (Dropout)         (None, 512)               0

 dense_23 (Dense)            (None, 1)                 513

=================================================================
Total params: 1639425 (6.25 MB)
Trainable params: 1639425 (6.25 MB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```





the model 3 yielded accuracy score of 0.7916

Summary

Out of 3 models, the model 2 significantly outperformed both model 1 and 2. However, it should be noted that the sample size for these models was quite small (60 each, 120 total) and may not yield the same result for the bigger sample set. It also seems indicative that loss/accuracy graph on validation set shows notable fluctuation which seems to indicate that our sample set wasn't big enough to reduce deviation on each sample. The first model seems to exhibit overfit past ~30 epochs, which indicates that batch size may have been too big for the sample size and the number of epochs should be reduced. Adding regularization for model 3 seems to have worsen the result by a lot. Perhaps given the sample size, losing any information significantly alters the prediction result of less frequent images in the sample.

For further analysis, increasing the sample size should be prioritized to generate more accurate and adaptable model. Once this is achieved, reducing the dimensionality of the model may help improving the result by silencing potential noises in the images via autoencoder or nmf. It would also be recommended to further tuning the model by varying optimizers and convolution layers as well.