Members name	Members id	Team number
اسلام حسن محمد عفيفي	201900136	3
ادهم سيد محمد توفيق	201900117	3
احمد ناصر عبدالستار احمد	201900107	3
يارا محمد حسن علي	201900954	3
سهيلة حسن حسني احمد	201900356	3
سيف الدين احمد سيف	201900362	3

Paper details :

Link : https://iopscience.iop.org/article/10.1088/1742-6596/1447/1/012021/pdf

Paper name : Multi-class Image Classification Using Deep Learning Algorithm

publisher name : W A Ezat , M M Dessouky and N A Ismail

year of publication : 2021

The dataset used : For the task of training and validation for classification considered four classes from the PASCAL VOC 2007 data-set. Many computer researchers and scientific papers use PASCAL 2007 as data-set for training and validation. PASCAL 2007 data-set is small-scale with 20 object classes. The PASCAL 2007 data-set has text files label every image as the image name and image class. All images are JPG format

Results : The following experimental results are obtained by deep learning framework CAFFE. The model has been trained and accuracy from the three training cycles is the accuracy of (87 %, 93 %, and 92.7 %) with an average accuracy value of 90.9 %

General Information on the selected dataset:

Link of dataset: https://www.kaggle.com/datasets/ayushv322/animal-classification

total number of samples : 4000 images

number of classes and their labels: four classes [Zebra _ elephant _ Rhino _ Buffalo]

Implementation details:

A) A CSV file has been created to contain an index for all images from classes [Zebra_elephant_bufflao_Rhino]

B) With python scripting code three CSV files have been generated to index the chosen images and partition images randomly into four equal subset [G1 – G2 – G3 - test] contain the images name, image classes

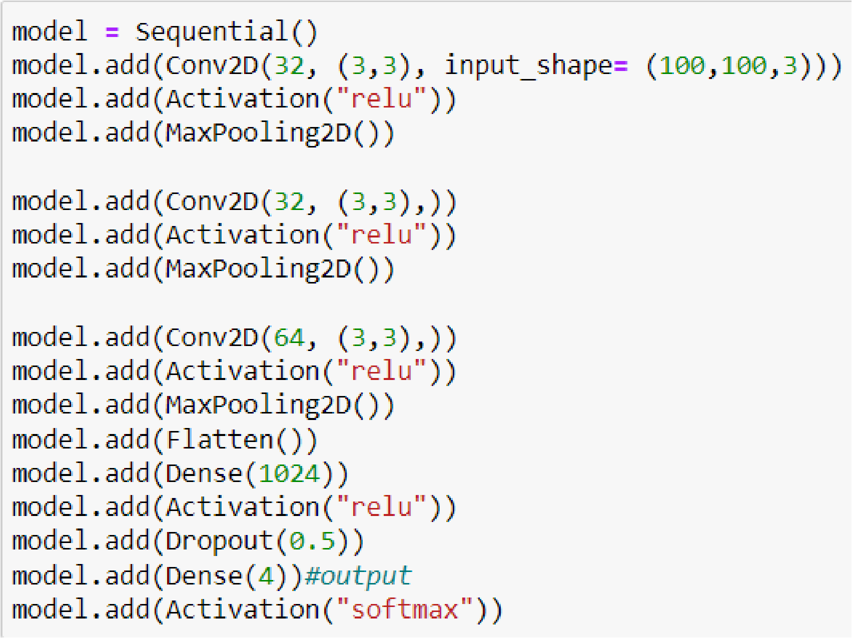
C) Each subset contain equal number of images (250 image for each class) and the total number is 1000.

D) The cross-validation technique has been used in training and validation. In the first training and validation cycle, the subsets G1 and G2 have been used for training the model and the subset G3 has been used for the validation

E) In the second training and validation cycle subsets G1 and G3 have been used for training and the subset G2 are used for the validation process. In the third training and validation cycle, subsets G2 and G3 have been used for the training process and subset G1 for the validation process.

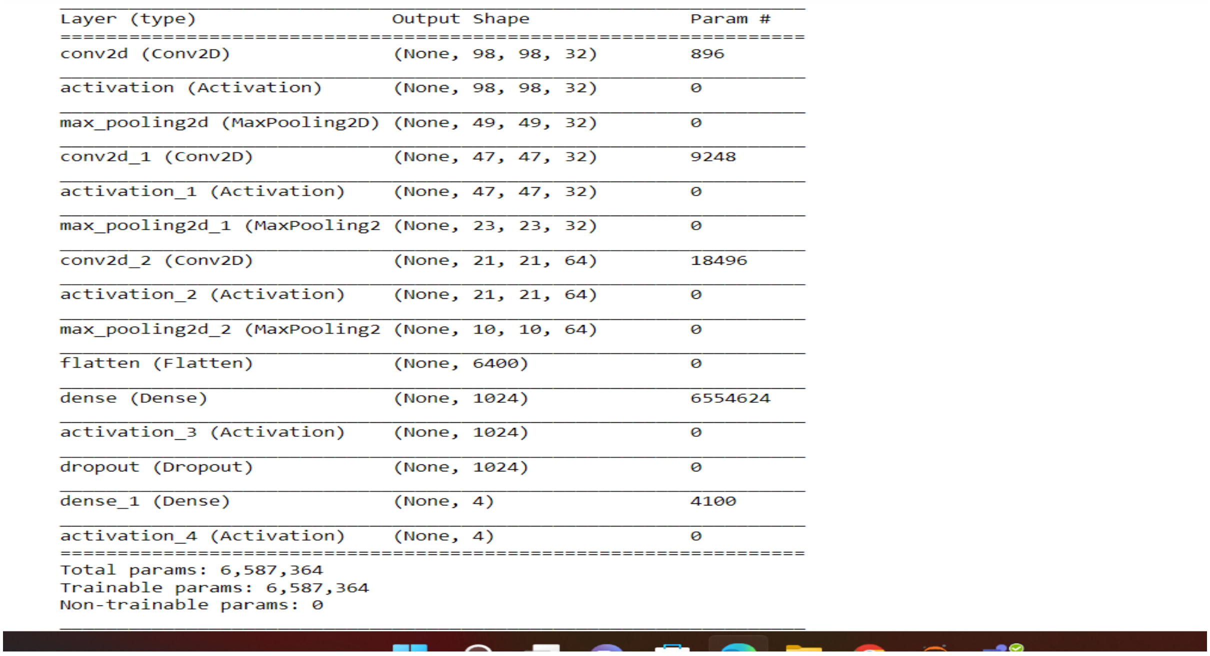
After each cycle we test the algorithm with (test data) and see the accuracy and try to develop it.

Architecture of model



```
model = Sequential()
model.add(Conv2D(32, (3,3), input_shape= (100,100,3)))
model.add(Activation("relu"))
model.add(MaxPooling2D())

model.add(Conv2D(32, (3,3),))
model.add(Activation("relu"))
model.add(MaxPooling2D())

model.add(Conv2D(64, (3,3),))
model.add(Activation("relu"))
model.add(MaxPooling2D())
model.add(Flatten())
model.add(Dense(1024))
model.add(Activation("relu"))
model.add(Dropout(0.5))
model.add(Dense(4))#output
model.add(Activation("softmax"))
```

Summary

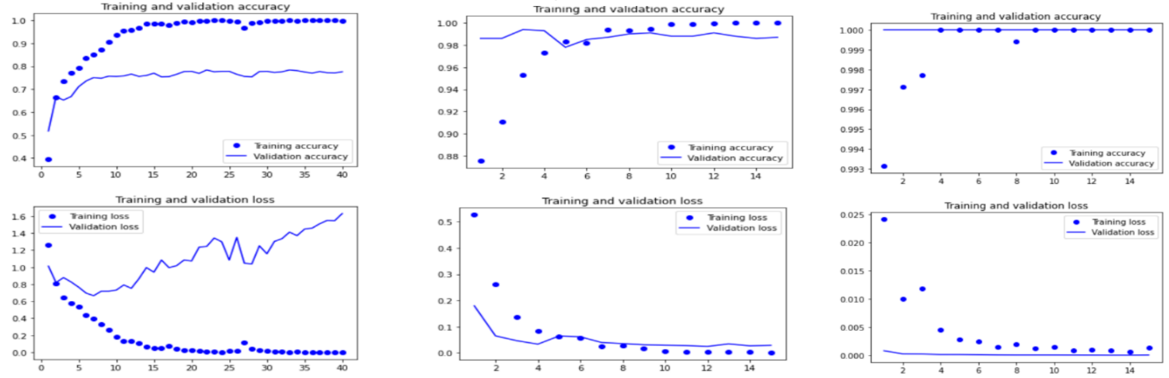


```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 98, 98, 32)        896
_________________________________________________________________
activation (Activation)      (None, 98, 98, 32)        0
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 49, 49, 32)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 47, 47, 32)        9248
_________________________________________________________________
activation_1 (Activation)    (None, 47, 47, 32)        0
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 23, 23, 32)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 21, 21, 64)        18496
_________________________________________________________________
activation_2 (Activation)    (None, 21, 21, 64)        0
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 10, 10, 64)        0
_________________________________________________________________
flatten (Flatten)            (None, 6400)              0
_________________________________________________________________
dense (Dense)                (None, 1024)              6554624
_________________________________________________________________
activation_3 (Activation)    (None, 1024)              0
_________________________________________________________________
dropout (Dropout)            (None, 1024)              0
_________________________________________________________________
dense_1 (Dense)              (None, 4)                 4100
_________________________________________________________________
activation_4 (Activation)    (None, 4)                 0
=================================================================
Total params: 6,587,364
Trainable params: 6,587,364
Non-trainable params: 0
_________________________________________________________________
```

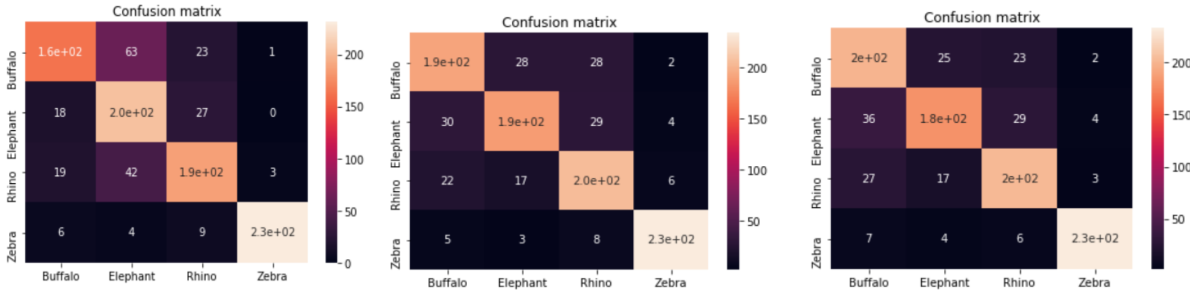
Results and visualizations

Accuracy on test images after cycle 1 =79% ,cycle 2 =81% ,cycle 3 = 81%
Accuracy average is = 80%

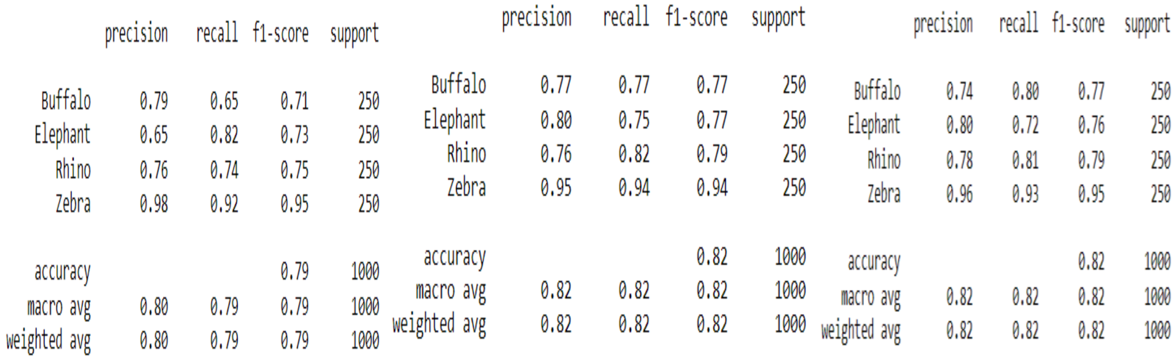
Train and validation (loss , accuracy) :



Confusion matrix on test images:



Reports on test images:

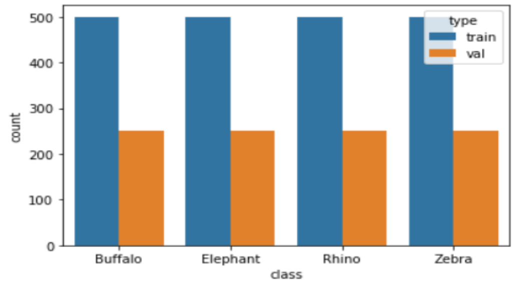


```
              precision    recall  f1-score   support

     Buffalo       0.79      0.65      0.71       250
    Elephant       0.65      0.82      0.73       250
       Rhino       0.76      0.74      0.75       250
       Zebra       0.98      0.92      0.95       250

    accuracy                           0.79      1000
   macro avg       0.80      0.79      0.79      1000
weighted avg       0.80      0.79      0.79      1000
```

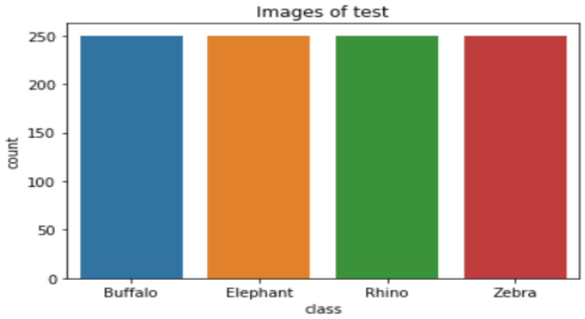
```
              precision    recall  f1-score   support

     Buffalo       0.77      0.77      0.77       250
    Elephant       0.80      0.75      0.77       250
       Rhino       0.76      0.82      0.79       250
       Zebra       0.95      0.94      0.94       250

    accuracy                           0.82      1000
   macro avg       0.82      0.82      0.82      1000
weighted avg       0.82      0.82      0.82      1000
```

```
              precision    recall  f1-score   support

     Buffalo       0.74      0.80      0.77       250
    Elephant       0.80      0.72      0.76       250
       Rhino       0.78      0.81      0.79       250
       Zebra       0.96      0.93      0.95       250

    accuracy                           0.82      1000
   macro avg       0.82      0.82      0.82      1000
weighted avg       0.82      0.82      0.82      1000
```

Images of train and validation:



images of test :


Images of test
count
Buffalo
Elephant
Rhino
Zebra
class