# ***Report***

**Team name =** Team 14

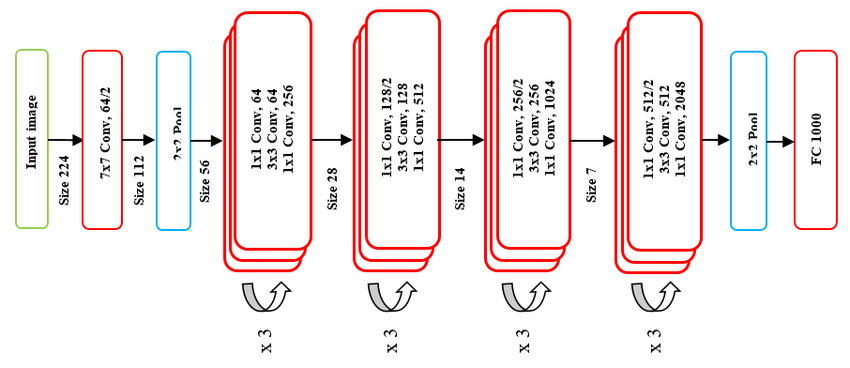
1. Hassan Elsayed Mohamed

# **Problems and Data**

This dataset contains natural images of size 150x150 for 6 categories (buildings, forest, glacier, mountain, sea, and street). The training set contains 6 folders, around 2300 for each category. The testing set contains 3050 images. The goal is to build a model that can recognize the category of each image in the testing set.

**The models we used**

1. **ResNet50**
2. **EfficientNetV2L**
3. **InceptionV3**
4. **Sequential**

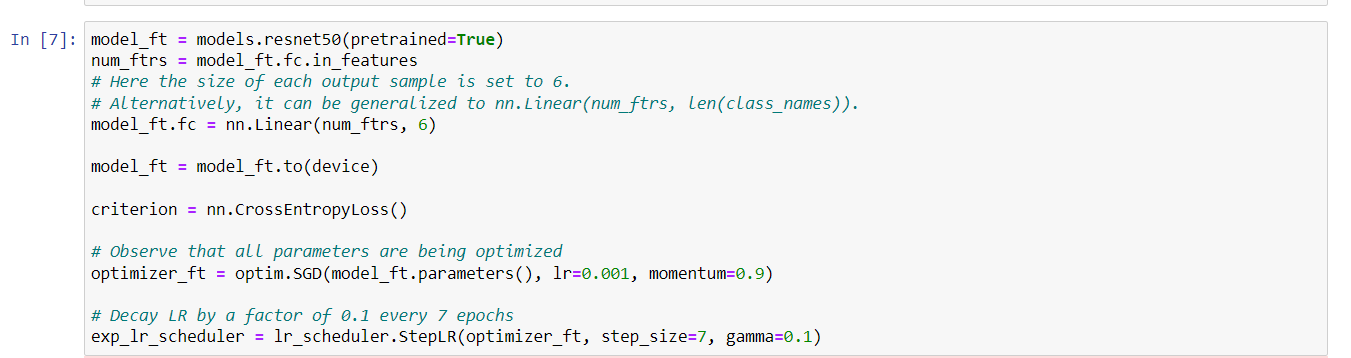
**First ResNet50**

We used pytorch to build and training the model.

Firstly, we are splitting train data into 80% training and 20% validation then make some transformation on it.

Secondly, we are using **SGD** optimizer with learning rate = 0.001, momentum=0.9 and observe that all parameters are being optimized,

Then decay LR by a factor of 0.1 every 7 epochs.

Finally, we got accuracy 93% in Validationand **0.92131** in the Testing

**Second EfficientNetV2L**

We used TensorFlow to build and training the model.

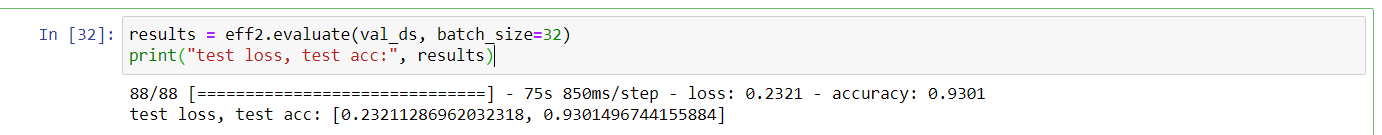
Firstly, we are splitting train data into 80% training and 20% validation.

Secondly, we are using **Adam** optimizer with learning rate = 0.001,

loss = sparse\_categorical\_crossentropy and Activation function = Rule and on the layer and last layer we used Softmax

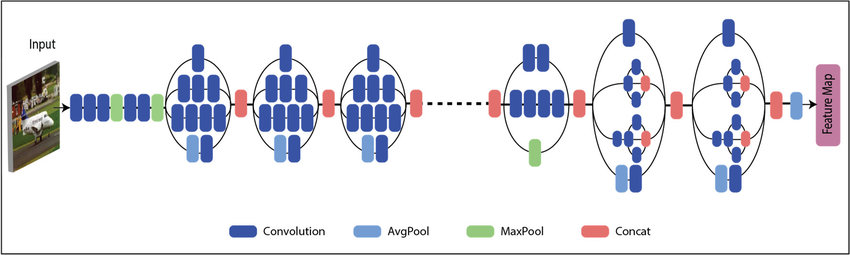
Finally, we got accuracy 0.9301 in Validationand **0.91896** in the Testing

**Model summary**

**Table

Description automatically generated with medium confidence**

**Third InceptionV3**

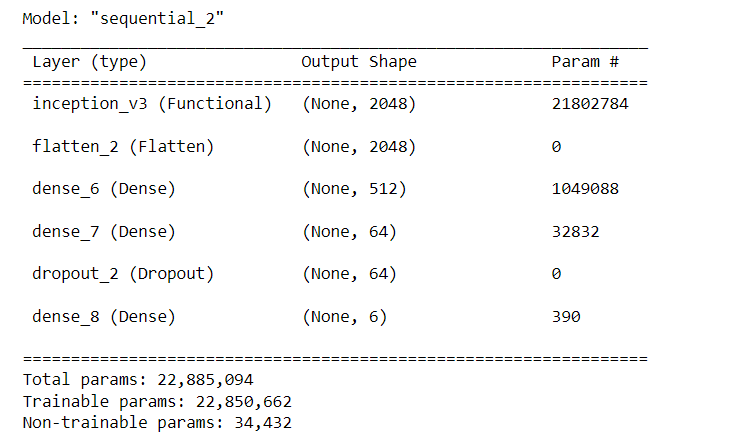
****

We used TensorFlow to build and training the model.

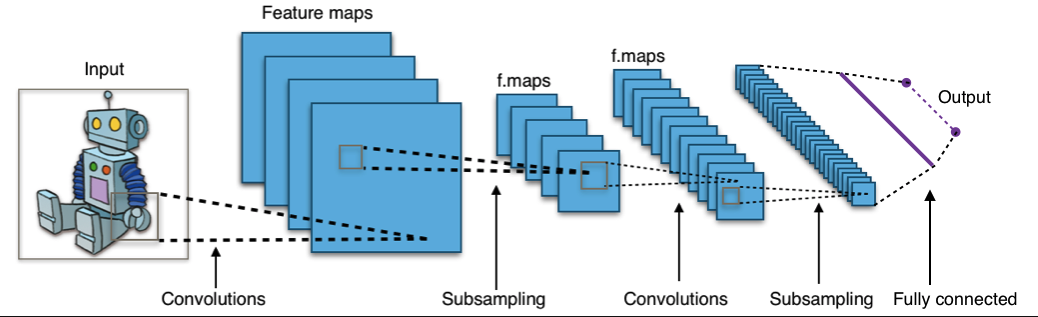
Firstly, we are splitting train data into 80% training and 20% validation.

Secondly, we are using **Adam** optimizer with learning rate = 0.001 and loss = categorical\_crossentropy, we do layer.trainable=True

Finally, we got accuracy 0.8513 in Validationand **0.86323** in the Testing

**Model summary**

**Fourth Sequential**



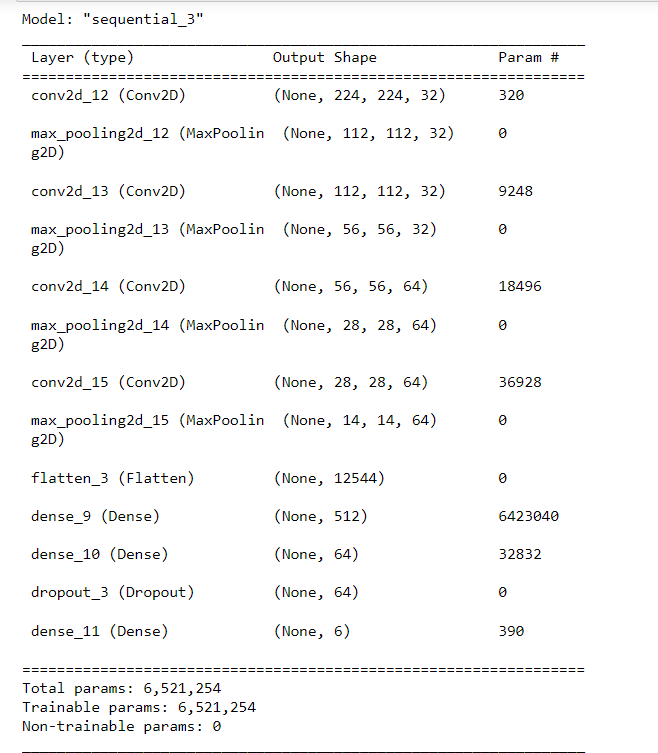
We used TensorFlow to build and training the model.

Firstly, we are splitting train data into 80% training and 20% validation.

Secondly, we are using **Adam** optimizer with learning rate = 0.001,

loss = sparse\_categorical\_crossentropy and Activation function = Rule and on the layer and last layer we used Softmax

Finally, we got accuracy 70.42% in Validation **it make overfiting**

**Model summary**