Presentation : CSE465

Semester: **Summer23**

Section: **03**

Faculty: **Azk**

Project Title: Comparative Analysis of Custom CNN and Feedforward Network for Tiny ImageNet Classification

Name	ID
Anik Roy Pranto	1921219042
Md. Raihan Uddin Sarker	1912115042
Momena Farhana Mim	1921873042
Sadia Ishrat Anika	1921115642

Date: Oct 30, 2023

About our Project

TOPIC: Comparative Study of Custom CNN vs. Feedforward Network for Image Classification using a small dataset from Tiny ImageNet

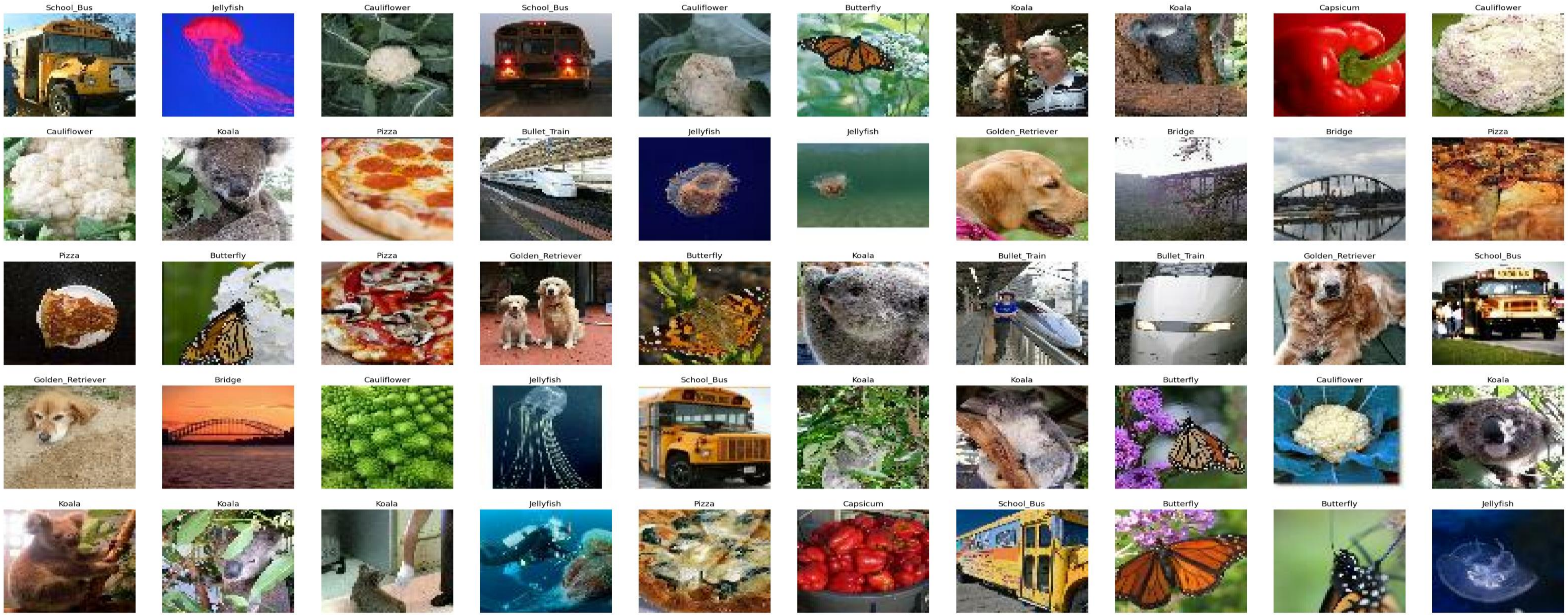
What is the Project About?:

- Comparison of two neural network architectures: Custom CNN and Feedforward Network
- Image classification using a small dataset from Tiny ImageNet (1000 images)
- Ten distinct categories for evaluation
- Assessment of model accuracy and efficiency(Regularization and Optimization)

Dataset statistics

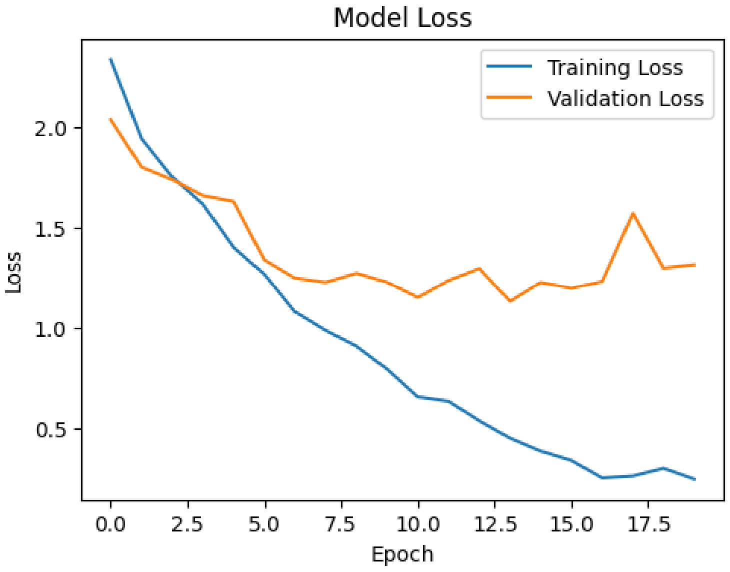
Dataset	Size	Number of Classes	Number of Samples		
			Train	Test	Validation
Tiny ImageNet	64x64	10	800	100	100

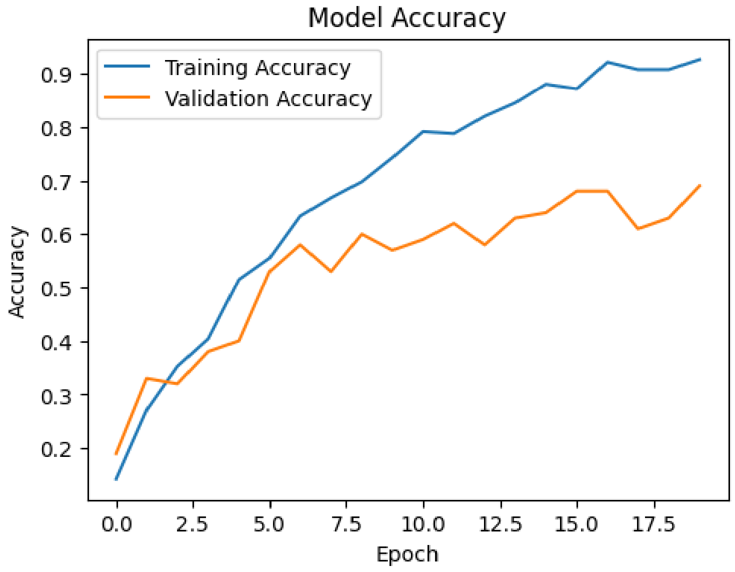
Snapshot of the Dataset:-



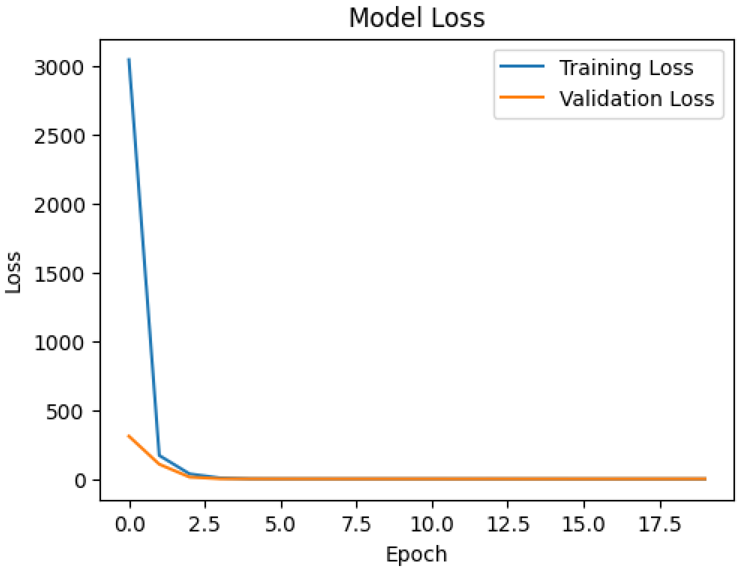
The Training curve *(learning and validation)*

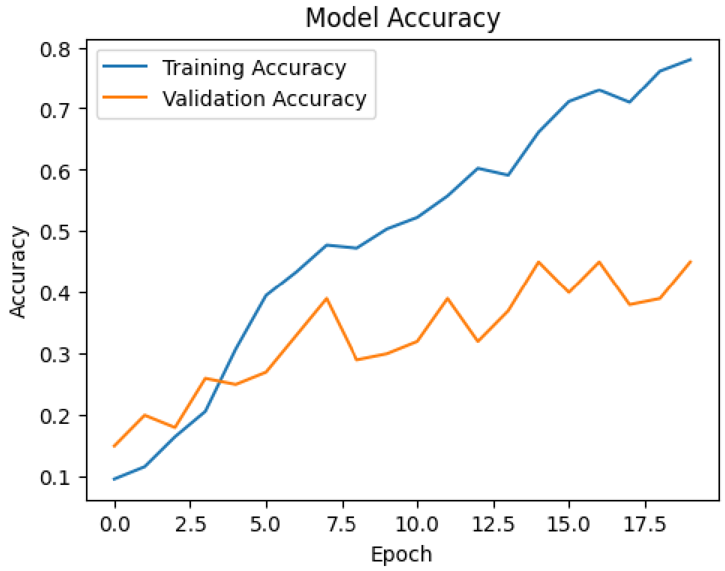
CNN Model:-





Feedforward Network:-

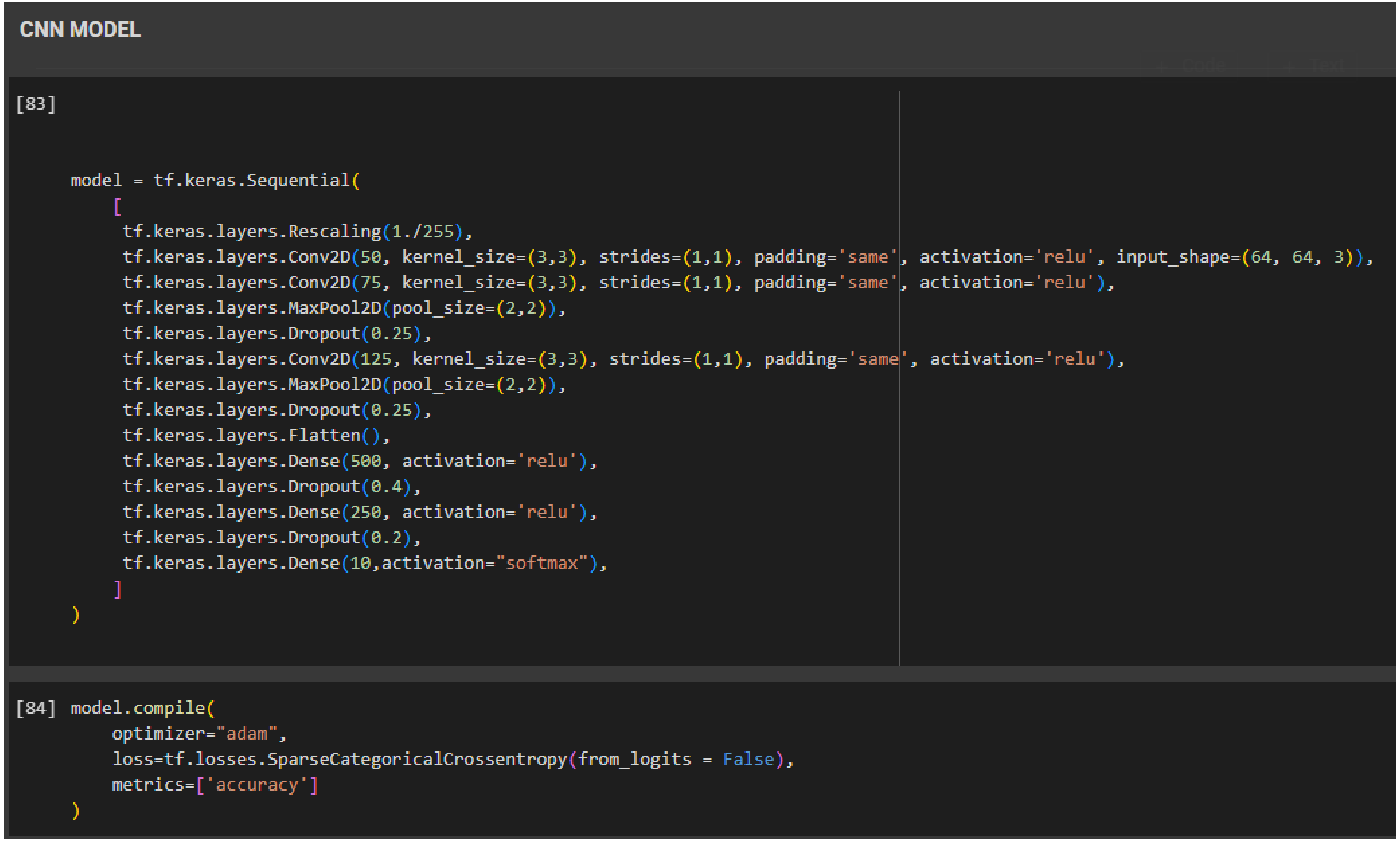




Statistics

Model Name	Required time to train	Average time spent on each epoch
CNN Model	Total 20 Epochs: (*59+57+56+56+59+60+57+56+57+55+58+54+57+58+59+57+57+58+56+56*)s = 1142s = **19.03m**	(*3+4+4+4+4+4+3+3+4+3+4+3+4+4+4+3+4+4+3+4*)s / 20 = 73s/20 = **3.65s**
Feedforward Network	Total 20 Epochs: *30s + 27s + 24s + 26s + 26s + 25s + 24s + 27s +25s + 24s+ 24s + 28s + 26s + 24s + 26s + 26s + 23s + 25s + 26s + 24s* = **8.14m**	(*2+2+1+2+2+2+2+2+2+1+1+2+2+2+2+2+1+2+2+1*)s /20 = **1.45s**

Regularization and Optimization used



CNN MODEL

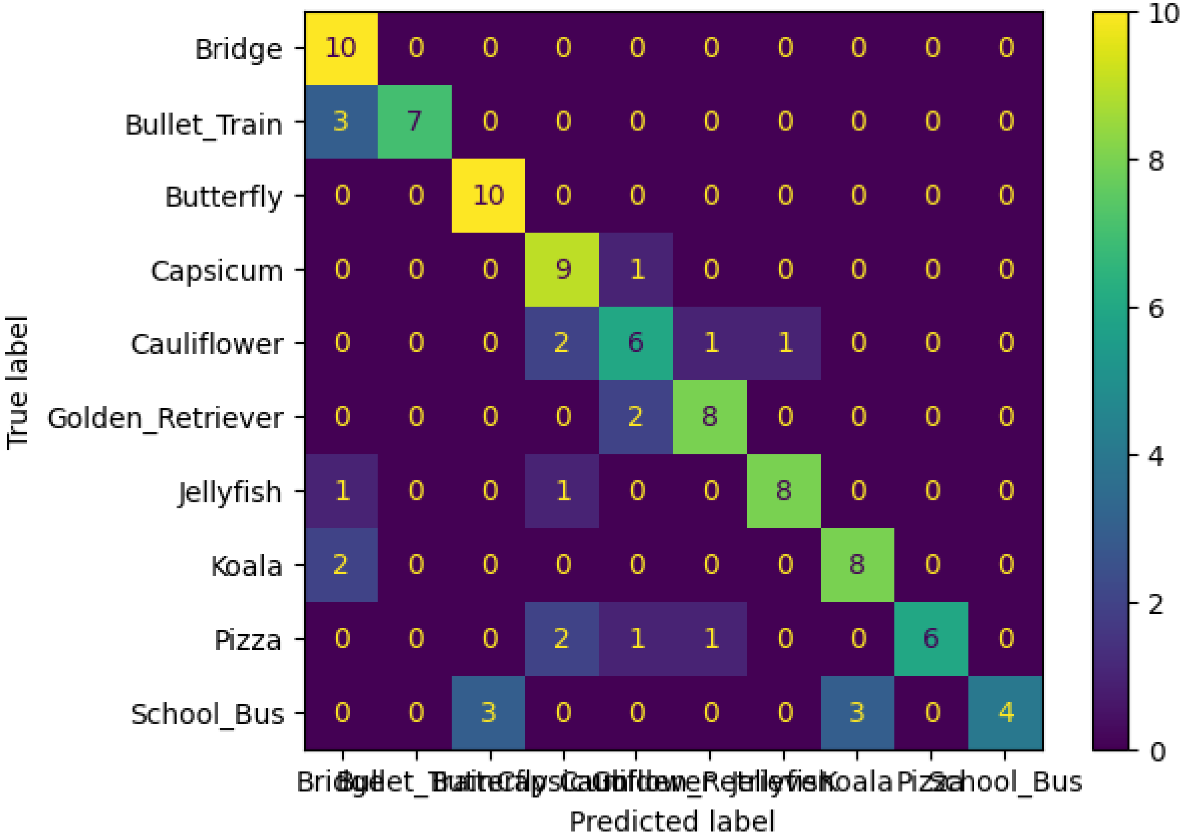
```
[83]


     model = tf.keras.Sequential(
         [
          tf.keras.layers.Rescaling(1./255),
          tf.keras.layers.Conv2D(50, kernel_size=(3,3), strides=(1,1), padding='same', activation='relu', input_shape=(64, 64, 3)),
          tf.keras.layers.Conv2D(75, kernel_size=(3,3), strides=(1,1), padding='same', activation='relu'),
          tf.keras.layers.MaxPool2D(pool_size=(2,2)),
          tf.keras.layers.Dropout(0.25),
          tf.keras.layers.Conv2D(125, kernel_size=(3,3), strides=(1,1), padding='same', activation='relu'),
          tf.keras.layers.MaxPool2D(pool_size=(2,2)),
          tf.keras.layers.Dropout(0.25),
          tf.keras.layers.Flatten(),
          tf.keras.layers.Dense(500, activation='relu'),
          tf.keras.layers.Dropout(0.4),
          tf.keras.layers.Dense(250, activation='relu'),
          tf.keras.layers.Dropout(0.2),
          tf.keras.layers.Dense(10,activation="softmax"),
         ]
     )
```

```
[84] model.compile(
         optimizer="adam",
         loss=tf.losses.SparseCategoricalCrossentropy(from_logits = False),
         metrics=['accuracy']
     )
```

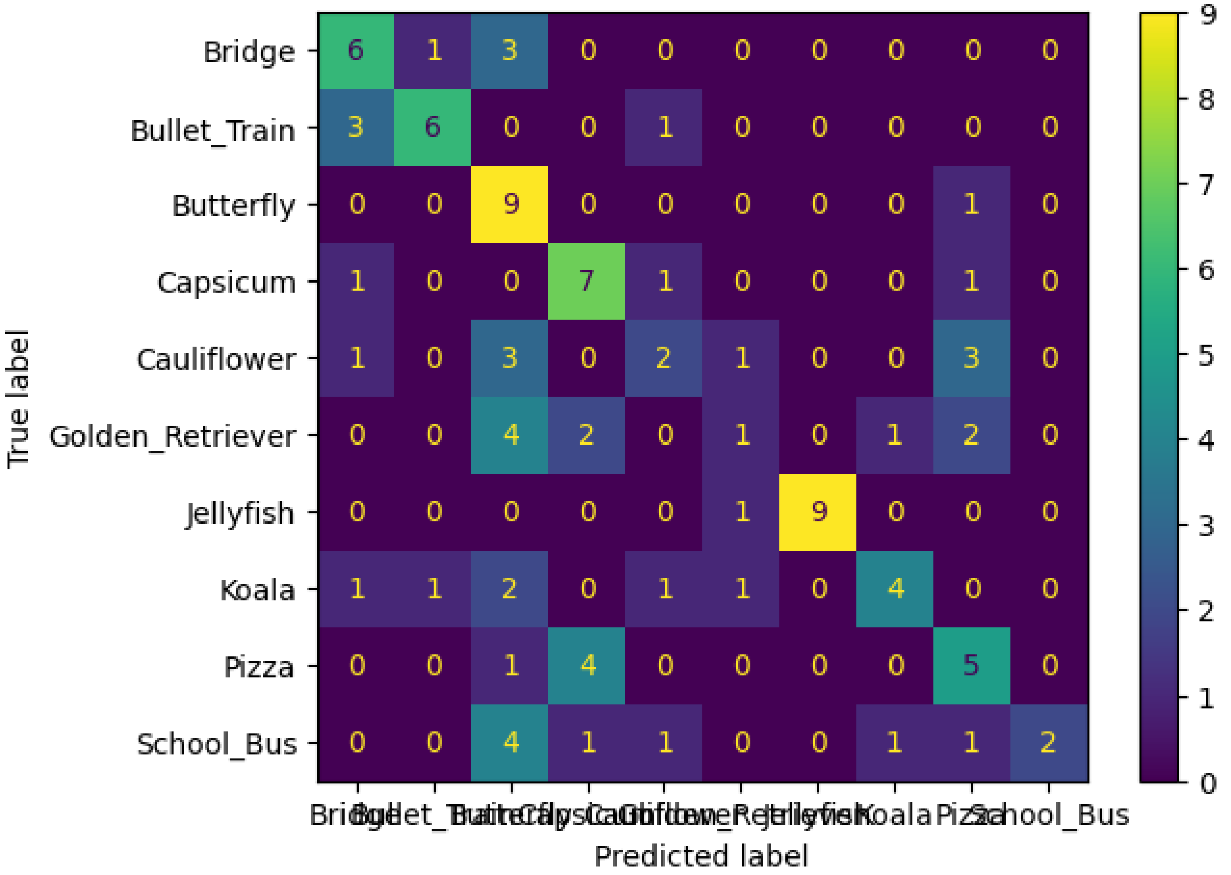
Confusion matrix

Confusion Matrix (CNN):



	Bridge	Bullet_Train	Butterfly	Capsicum	Cauliflower	Golden_Retriever	Jellyfish	Koala	Pizza	School_Bus
Bridge	10	0	0	0	0	0	0	0	0	0
Bullet_Train	3	7	0	0	0	0	0	0	0	0
Butterfly	0	0	10	0	0	0	0	0	0	0
Capsicum	0	0	0	9	1	0	0	0	0	0
Cauliflower	0	0	0	2	6	1	1	0	0	0
Golden_Retriever	0	0	0	0	2	8	0	0	0	0
Jellyfish	1	0	0	1	0	0	8	0	0	0
Koala	2	0	0	0	0	0	0	8	0	0
Pizza	0	0	0	2	1	1	0	0	6	0
School_Bus	0	0	3	0	0	0	0	3	0	4

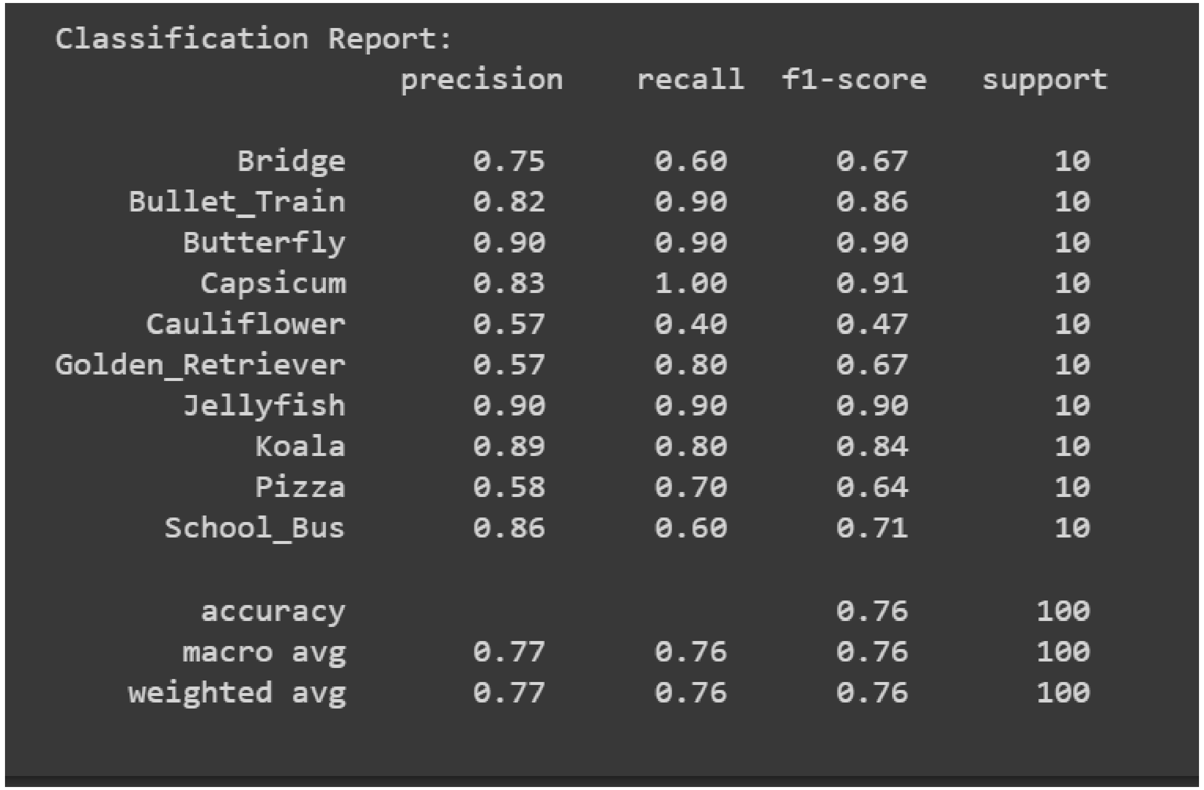
Confusion Matrix (Feedforward):



	Bridge	Bullet_Train	Butterfly	Capsicum	Cauliflower	Golden_Retriever	Jellyfish	Koala	Pizza	School_Bus
Bridge	6	1	3	0	0	0	0	0	0	0
Bullet_Train	3	6	0	0	1	0	0	0	0	0
Butterfly	0	0	9	0	0	0	0	0	1	0
Capsicum	1	0	0	7	1	0	0	0	1	0
Cauliflower	1	0	3	0	2	1	0	0	3	0
Golden_Retriever	0	0	4	2	0	1	0	1	2	0
Jellyfish	0	0	0	0	0	1	9	0	0	0
Koala	1	1	2	0	1	1	0	4	0	0
Pizza	0	0	1	4	0	0	0	0	5	0
School_Bus	0	0	4	1	1	0	0	1	1	2

Classification Report

CNN

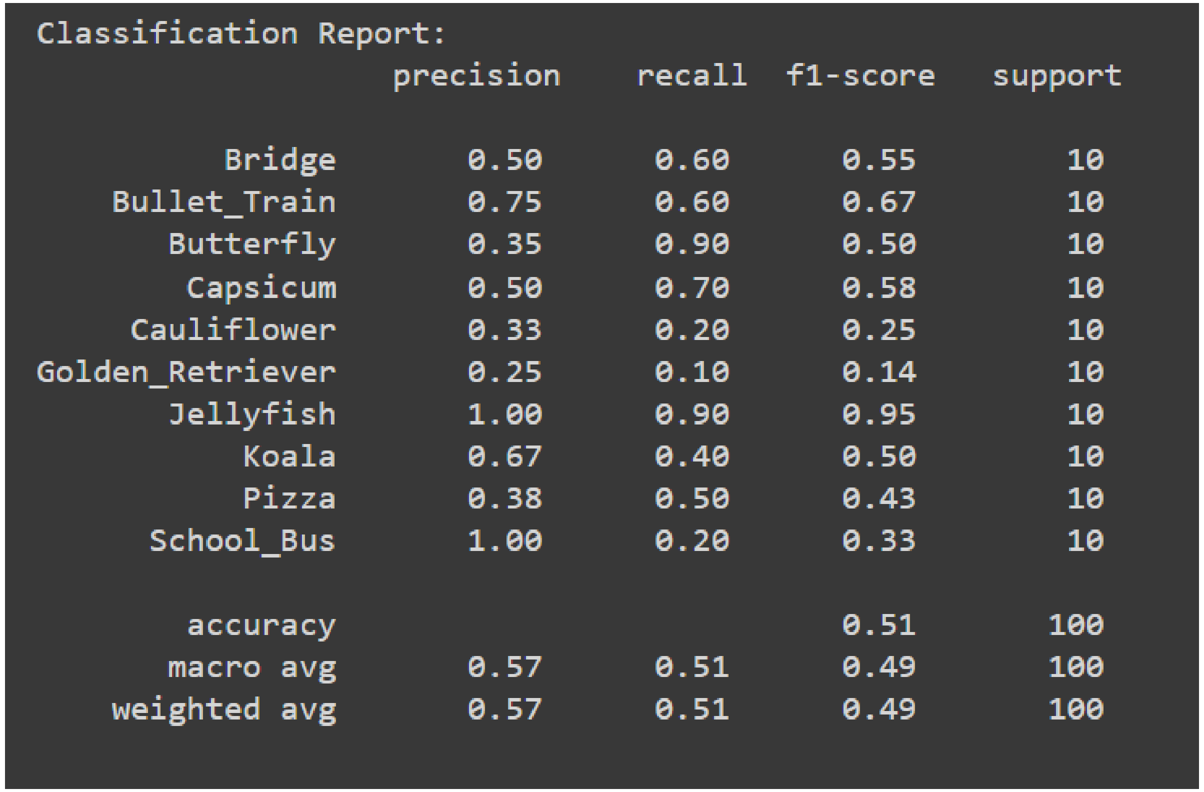


```
Classification Report:
                    precision    recall  f1-score   support

          Bridge       0.75      0.60      0.67        10
    Bullet_Train       0.82      0.90      0.86        10
       Butterfly       0.90      0.90      0.90        10
        Capsicum       0.83      1.00      0.91        10
     Cauliflower       0.57      0.40      0.47        10
 Golden_Retriever       0.57      0.80      0.67        10
       Jellyfish       0.90      0.90      0.90        10
           Koala       0.89      0.80      0.84        10
           Pizza       0.58      0.70      0.64        10
      School_Bus       0.86      0.60      0.71        10

        accuracy                           0.76       100
       macro avg       0.77      0.76      0.76       100
    weighted avg       0.77      0.76      0.76       100
```

FFN



```
Classification Report:
                    precision    recall  f1-score   support

          Bridge       0.50      0.60      0.55        10
    Bullet_Train       0.75      0.60      0.67        10
       Butterfly       0.35      0.90      0.50        10
        Capsicum       0.50      0.70      0.58        10
     Cauliflower       0.33      0.20      0.25        10
Golden_Retriever       0.25      0.10      0.14        10
       Jellyfish       1.00      0.90      0.95        10
           Koala       0.67      0.40      0.50        10
           Pizza       0.38      0.50      0.43        10
      School_Bus       1.00      0.20      0.33        10

        accuracy                           0.51       100
       macro avg       0.57      0.51      0.49       100
    weighted avg       0.57      0.51      0.49       100
```

Thank You~