

```
In [1]: # a. Load libraries
import tensorflow as tf
from tensorflow.keras import layers, models

from tensorflow.keras.applications import MobileNetV2
```

```
In [2]: # Load an existing dataset
(x_train, y_train), (x_test, y_test) = tf.keras.datasets.cifar10.load_data()
```

```
In [3]: # Normalize values
x_train = x_train / 255.0
x_test = x_test / 255.0
```

```
In [4]: # b. Load a pre-trained CNN model
base_model = MobileNetV2(
    weights='imagenet',
    include_top=False,
    input_shape=(96, 96, 3)
)
```

```
In [5]: # Resize CIFAR-10 images to MobileNet size
x_train = tf.image.resize(x_train, (96, 96))
x_test = tf.image.resize(x_test, (96, 96))
```

```
In [6]: # Freeze Lower Layers
base_model.trainable = False
```

```
In [7]: model = models.Sequential([
    base_model,
    layers.Flatten(),
    layers.Dense(128, activation='relu'),
    layers.Dropout(0.3),
    layers.Dense(10, activation='softmax')
])
```

```
In [8]: # d. Compile and train classifier Layers
model.compile(optimizer='adam',
              loss='sparse_categorical_crossentropy',
              metrics=['accuracy'])
```

```
In [9]: model.fit(x_train, y_train, epochs=3, validation_split=0.2)

Epoch 1/3
1250/1250 ————— 400s 307ms/step - accuracy: 0.6427 - loss: 1.0631 - val_accuracy: 0.7651 - val_loss: 0.6910
Epoch 2/3
1250/1250 ————— 343s 274ms/step - accuracy: 0.7304 - loss: 0.7737 - val_accuracy: 0.7903 - val_loss: 0.6199
Epoch 3/3
1250/1250 ————— 355s 284ms/step - accuracy: 0.7566 - loss: 0.6939 - val_accuracy: 0.7867 - val_loss: 0.6385

Out[9]: <keras.src.callbacks.history.History at 0x1f5d3770380>
```

```
In [10]: # e. Fine-tune: unfreeze some of the deeper Layers
base_model.trainable = True
```

```
In [11]: model.compile(optimizer='adam',
                      loss='sparse_categorical_crossentropy',
                      metrics=['accuracy'])
```

```
In [12]: model.fit(x_train, y_train, epochs=2, validation_split=0.2)

Epoch 1/2
1250/1250 ————— 1157s 911ms/step - accuracy: 0.7040 - loss: 0.9809 - val_accuracy: 0.3683 - val_loss: 5.2608
Epoch 2/2
1250/1250 ————— 1159s 927ms/step - accuracy: 0.8114 - loss: 0.6259 - val_accuracy: 0.4126 - val_loss: 5.0996

Out[12]: <keras.src.callbacks.history.History at 0x1f5d30bce90>
```

```
In [13]: test_loss, test_acc = model.evaluate(x_test, y_test)
print("Test Accuracy:", test_acc)

313/313 ————— 89s 283ms/step - accuracy: 0.4075 - loss: 5.1032
Test Accuracy: 0.4074999988079071
```

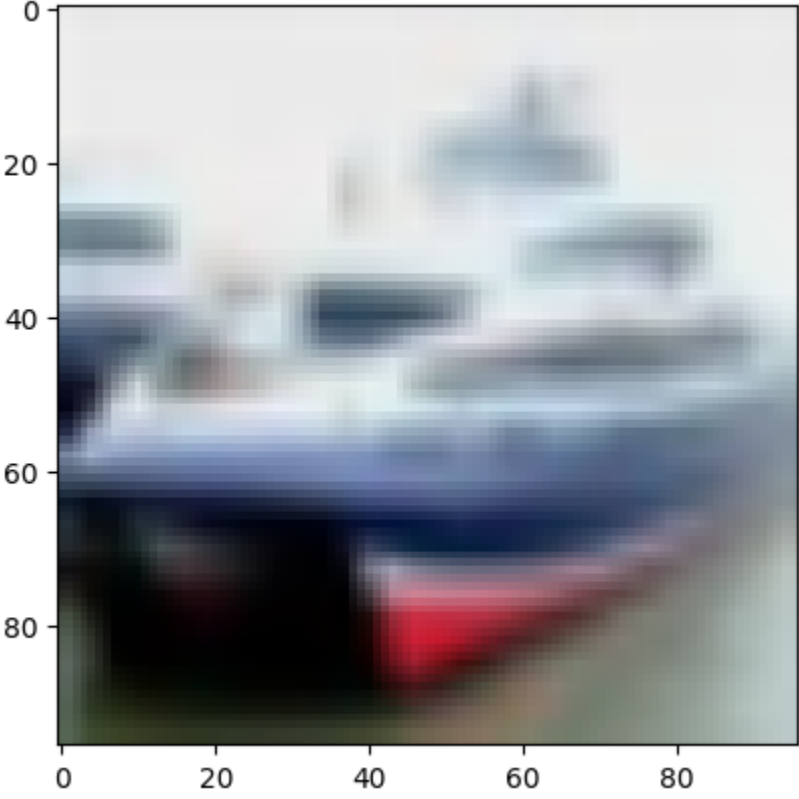
```
In [14]: import numpy as np
from tensorflow.keras.preprocessing import image
```

```
In [18]: class_names = ["airplane", "car", "bird", "cat", "deer",
                        "dog", "frog", "horse", "ship", "truck"]
```

```
In [2]: import matplotlib.pyplot as plt
```

```
In [22]: plt.imshow(x_test[1])

Out[22]: <matplotlib.image.AxesImage at 0x1f5e835ba70>
```



```
In [24]: testpred=model.predict(x_test)
print(testpred[1])

313/313 ————— 106s 313ms/step
[3.2820227e-04 1.6688218e-04 1.1158143e-13 3.9202867e-13 1.6779693e-19
 5.4099306e-16 1.1592558e-22 3.2431855e-23 9.9950469e-01 1.2320741e-07]
```

