1. WORKING OF CNN ARCHITECTURE TO CLASSIFY IMAGES

EX.N0 : 1	**WORKING OF CNN ARCHITECTURE TO CLASSIFY IMAGES**
DATE : 21/01/2025	

AIM:

To implement and demonstrate a Convolutional Neural Network (CNN) for image classification using the CIFAR-10 dataset in TensorFlow and Keras.

ALGORITHM:

Step 1: Import necessary libraries.

Step 2: Load and pre-process the CIFAR-10 dataset.

Step 3: Build the CNN model using Keras Sequential API.

Step 4: Compile the model with appropriate loss function and optimizer.

Step 5: Train the model on the training dataset.

Step 6: Evaluate the model on the test dataset.

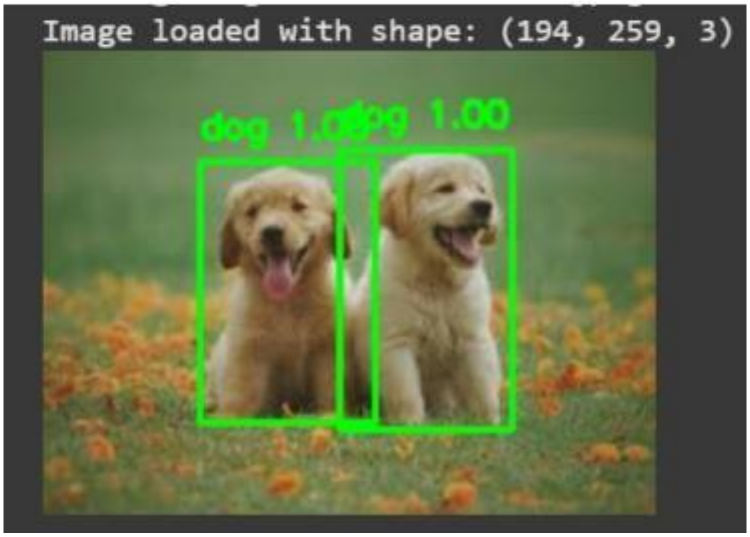
Step 7: Visualize a few predictions to validate performance.

PROGRAM:

```
import tensorflow as tf from tensorflow.keras
import layers, models import
matplotlib.pyplot as plt from
tensorflow.keras.datasets import cifar10
import numpy as np
(x_train, y_train), (x_test, y_test) = cifar10.load_data() x_train
= x_train.astype('float32') / 255.0
```

```
x_test = x_test.astype('float32') / 255.0 model = models.Sequential([
layers.Conv2D(32, (3, 3), activation='relu', input_shape=(32, 32,
3)), layers.MaxPooling2D((2, 2)), layers.Conv2D(64, (3, 3),
activation='relu'), layers.MaxPooling2D((2, 2)), layers.Conv2D(64,
(3, 3), activation='relu'), layers.Flatten(), layers.Dense(64,
activation='relu'), layers.Dense(10, activation='softmax')
])
model.compile(optimizer='adam',
loss='sparse_categorical_crossentropy', metrics=['accuracy'])
history = model.fit(x_train, y_train, epochs=10, batch_size=64,
validation_data=(x_test, y_test)) test_loss, test_acc =
model.evaluate(x_test, y_test, verbose=2) print(f'\nTest
accuracy: {test_acc:.4f}') class_names =
['airplane','automobile','bird','cat','deer',
'dog','frog','horse','ship','truck']
predictions = model.predict(x_test)
plt.figure(figsize=(10,5)) for i in
range(5):
plt.subplot(1, 5, i+1) plt.xticks([]) plt.yticks([])
plt.grid(False) plt.imshow(x_test[i]) pred_label =
class_names[np.argmax(predictions[i])] true_label =
class_names[y_test[i][0]] plt.xlabel(f"Pred:
{pred_label}\nTrue: {true_label}") plt.tight_layout()
plt.show()
```

OUTPUT:



```
Image loaded with shape: (194, 259, 3)
```

RESULT :

Thus, to implement and demonstrate a Convolutional Neural Network (CNN) for image classification using the CIFAR-10 dataset in TensorFlow and Keras.