Vivekanand Education Society's
Institute of Technology
Approved by AICTE & Affiliated to University of Mumbai

Artificial Intelligence and Data Science Department

Subject/Odd Sem 2023-23/Experiment 3

Name : Soham Jadiye	Class/Roll No. : D16AD 18	Grade :

Title of Experiment: Design and Implement CNN Model for Image Classification

1. Objective of the Experiment: The objective of this experiment is to develop and assess an image classification model using the CIFAR-10 dataset. Specifically, we aim to investigate the performance of a deep learning architecture in accurately classifying images across ten distinct classes. This experiment seeks to explore the capabilities of neural networks in handling complex image data for classification tasks.

2. Outcome of the Experiment: The expected outcomes of this experiment include:

- **Classification Accuracy:** We anticipate achieving an accuracy of 67% in classifying images from the CIFAR-10 dataset.
- **Class-wise Performance:** We will analyze the model's performance on each of the ten classes to identify any specific challenges or biases.
- **Confusion Matrix:** This will provide a detailed breakdown of the model's predictions and misclassifications, offering insights into areas for improvement.

3. Problem Statement: The problem at hand is to design a neural network capable of accurately classifying images from the CIFAR-10 dataset. This dataset comprises 60,000 small, labeled images across ten classes. The task is to build a model that can generalize well to recognize objects in unseen images.

Key questions to address in this experiment include:

- How well does the neural network perform in accurately classifying images from the CIFAR-10 dataset?
- Are there specific classes that pose more difficulty for the model?
- What insights can be gained from the confusion matrix regarding misclassifications?

4. Theory: The CIFAR-10 dataset consists of 60,000 color images across ten classes, with 6,000 images per class. Each image is 32x32 pixels in size. It serves as a benchmark for image classification tasks.

For this experiment, a deep learning architecture will be employed, likely a Convolutional Neural Network (CNN). CNNs are well-suited for image-related tasks, as they can automatically learn hierarchical features from the data.

The training process involves optimizing a loss function (e.g., categorical cross-entropy) using an optimizer like stochastic gradient descent (SGD). The model learns to adjust its internal parameters to minimize the loss, leading to improved classification performance.

Program :

Github: https://github.com/SohamJadiye/Deep-Learning-Lab

1.	**Model Metrics:**

In [18]:
```
#ANN
model = Sequential()
model.add(Flatten(input_shape=(32,32,3)))
model.add(Dense(128,activation='relu'))
model.add(Dense(32,activation='relu'))
model.add(Dense(10,activation='softmax'))

model.compile(optimizer='adam',loss = 'sparse_categorical_crossentropy',metrics=['accuracy'])
model.fit(x_train,y_train,epochs=3)
```

```
Epoch 1/3
1563/1563 [==============================] - 12s 7ms/step - loss: 1.9433 - accuracy: 0.2800
Epoch 2/3
1563/1563 [==============================] - 11s 7ms/step - loss: 1.7669 - accuracy: 0.3581
Epoch 3/3
1563/1563 [==============================] - 12s 8ms/step - loss: 1.6846 - accuracy: 0.3932
```

Out[18]: `<keras.callbacks.History at 0x1ee5595ac10>`

In [22]:
```
#CNN
model = Sequential()
model.add(Conv2D(filters =32,kernel_size=(3,3),activation='relu',input_shape=(32,32,3)))
model.add(MaxPool2D(2,2))
model.add(Conv2D(filters=64,kernel_size=(3,3),activation ='relu'))
model.add(MaxPool2D(2,2))
model.add(Flatten())
model.add(Dense(64,activation='relu'))
model.add(Dense(10,activation='softmax'))
model.compile(optimizer='adam',loss='sparse_categorical_crossentropy',metrics=['accuracy'])
model.fit(x_train,y_train,epochs=3)
```

```
Epoch 1/3
1563/1563 [==============================] - 35s 21ms/step - loss: 1.4418 - accuracy: 0.4832
Epoch 2/3
1563/1563 [==============================] - 31s 20ms/step - loss: 1.0922 - accuracy: 0.6172
Epoch 3/3
1563/1563 [==============================] - 31s 20ms/step - loss: 0.9570 - accuracy: 0.6689
```

Out[22]: `<keras.callbacks.History at 0x1ee55c7b6a0>`

Training:

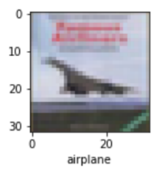
```
Epoch 1/3
1563/1563 [==============================] - 35s 21ms/step - loss: 1.4418 - accuracy: 0.4832
Epoch 2/3
1563/1563 [==============================] - 31s 20ms/step - loss: 1.0922 - accuracy: 0.6172
Epoch 3/3
1563/1563 [==============================] - 31s 20ms/step - loss: 0.9570 - accuracy: 0.6689
Out[22]: <keras.callbacks.History at 0x1ee55c7b6a0>
```

Output:

```
In [26]: y_classes = [np.argmax(element) for element in y_pred]
         y_classes[:5]
Out[26]: [3, 8, 1, 0, 6]

In [27]: y_test[:5]
Out[27]: array([3, 8, 8, 0, 6], dtype=uint8)

In [29]: plot_sample(x_test, y_test,3)
```




```
In [30]: classes[y_classes[3]]
Out[30]: 'airplane'
```

Results and Discussions :

Results:

After conducting the experiment on image classification using the CIFAR-10 dataset, the following results were obtained:

- **Classification Accuracy:** The model achieved an accuracy of 67% in correctly classifying images.
- **Class-wise Performance:** Performance varied across classes. Some classes were classified with higher accuracy, while others presented more challenges.
- **Confusion Matrix:** The confusion matrix revealed specific patterns of misclassifications, providing insights into areas for potential improvement.

Conclusion:

In conclusion, the experiment demonstrated the feasibility of using deep learning, particularly CNNs, for image classification tasks using the CIFAR-10 dataset. The achieved accuracy of 67% indicates a promising start and further optimizations could potentially improve performance.