Assignment No:10

Title:Python Program to implement CNN object detection. Discuss numerous performance evaluations.

Name: Tavhare Ruchita Sharad

Roll No:58

In [15]:
```
import keras
from keras.datasets import cifar10
from keras.models import Sequential
from keras.layers import Dense, Dropout, Flatten
from keras.layers import Conv2D, MaxPooling2D
from keras.optimizers import SGD
from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

In [17]:
```
# Load CIFAR-10 dataset
(X_train, y_train), (X_test, y_test) = cifar10.load_data()

# Define the model
model = Sequential()
model.add(Conv2D(32, (3, 3), activation='relu', input_shape=(32, 32, 3)))
model.add(Conv2D(32, (3, 3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))
model.add(Conv2D(64, (3, 3), activation='relu'))
model.add(Conv2D(64, (3, 3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))
model.add(Flatten())
model.add(Dense(512, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(10, activation='softmax'))
```

In [19]:
```
# Define data generators
train_datagen = ImageDataGenerator(rescale=1./255, shear_range=0.2, zoom_range=0.2, horizontal_flip=True)
test_datagen = ImageDataGenerator(rescale=1./255)
```

In [21]:
```
# Prepare the data
train_set = train_datagen.flow(X_train, y_train, batch_size=32)
test_set = test_datagen.flow(X_test, y_test, batch_size=32)
```

In [31]:
```
# Compile the model
sgd = SGD(learning_rate=0.01, momentum=0.9, nesterov=True)
model.compile(loss='sparse_categorical_crossentropy', optimizer=sgd, metrics=['accuracy'])
```

In [ ]:
```
# Train the model using fit method (replacing fit_generator)
model.fit(train_set, steps_per_epoch=len(X_train)//32, epochs=100, validation_data=test_set, validation_steps=l
```

```
Epoch 1/100
1562/1562 ━━━━━━━━━━━━━━━━━━━━ 37s 23ms/step - accuracy: 0.2558 - loss: 1.9764 - val_accuracy: 0.4716 - val_loss
: 1.4296
Epoch 2/100
   1/1562 ━━━━━━━━━━━━━━━━━━━━ 20s 13ms/step - accuracy: 0.4062 - loss: 1.3907
C:\Users\Admin\anaconda3\Lib\site-packages\keras\src\trainers\epoch_iterator.py:107: UserWarning: Your input ran
out of data; interrupting training. Make sure that your dataset or generator can generate at least `steps_per_ep
och * epochs` batches. You may need to use the `.repeat()` function when building your dataset.
  self._interrupted_warning()
```

```
1562/1562 ━━━━━━━━━━━━━━━━━━━━ 2s 1ms/step - accuracy: 0.4062 - loss: 1.3907 - val_accuracy: 0.4911 - val_loss:
1.3956
Epoch 3/100
1562/1562 ━━━━━━━━━━━━━━━━━━━━ 36s 23ms/step - accuracy: 0.4516 - loss: 1.5001 - val_accuracy: 0.5387 - val_loss
: 1.2710
Epoch 4/100
1562/1562 ━━━━━━━━━━━━━━━━━━━━ 2s 1ms/step - accuracy: 0.4062 - loss: 1.2545 - val_accuracy: 0.5406 - val_loss:
1.2664
Epoch 5/100
1562/1562 ━━━━━━━━━━━━━━━━━━━━ 36s 23ms/step - accuracy: 0.5150 - loss: 1.3523 - val_accuracy: 0.5720 - val_loss
: 1.1871
Epoch 6/100
1562/1562 ━━━━━━━━━━━━━━━━━━━━ 2s 1ms/step - accuracy: 0.4688 - loss: 1.4107 - val_accuracy: 0.5744 - val_loss:
1.1847
Epoch 7/100
1562/1562 ━━━━━━━━━━━━━━━━━━━━ 36s 23ms/step - accuracy: 0.5556 - loss: 1.2463 - val_accuracy: 0.6451 - val_loss
: 1.0050
Epoch 8/100
1562/1562 ━━━━━━━━━━━━━━━━━━━━ 2s 1ms/step - accuracy: 0.5625 - loss: 1.0499 - val_accuracy: 0.6468 - val_loss:
1.0023
Epoch 9/100
1562/1562 ━━━━━━━━━━━━━━━━━━━━ 36s 23ms/step - accuracy: 0.5862 - loss: 1.1641 - val_accuracy: 0.6581 - val_loss
: 0.9714
Epoch 10/100
1562/1562 ━━━━━━━━━━━━━━━━━━━━ 2s 1ms/step - accuracy: 0.3750 - loss: 1.2052 - val_accuracy: 0.6589 - val_loss:
0.9623
Epoch 11/100
1562/1562 ━━━━━━━━━━━━━━━━━━━━ 36s 23ms/step - accuracy: 0.6138 - loss: 1.0925 - val_accuracy: 0.6686 - val_loss
: 0.9505
Epoch 12/100
1562/1562 ━━━━━━━━━━━━━━━━━━━━ 2s 1ms/step - accuracy: 0.5312 - loss: 1.4861 - val_accuracy: 0.6781 - val_loss:
0.9249
Epoch 13/100
1562/1562 ━━━━━━━━━━━━━━━━━━━━ 36s 23ms/step - accuracy: 0.6335 - loss: 1.0496 - val_accuracy: 0.6822 - val_loss
: 0.8944
Epoch 14/100
1562/1562 ━━━━━━━━━━━━━━━━━━━━ 2s 1ms/step - accuracy: 0.6562 - loss: 1.1988 - val_accuracy: 0.6866 - val_loss:
0.8852
Epoch 15/100
1562/1562 ━━━━━━━━━━━━━━━━━━━━ 36s 23ms/step - accuracy: 0.6380 - loss: 1.0280 - val_accuracy: 0.6959 - val_loss
: 0.8776
Epoch 16/100
1562/1562 ━━━━━━━━━━━━━━━━━━━━ 2s 1ms/step - accuracy: 0.6875 - loss: 0.9077 - val_accuracy: 0.6917 - val_loss:
0.8862
Epoch 17/100
1562/1562 ━━━━━━━━━━━━━━━━━━━━ 36s 23ms/step - accuracy: 0.6558 - loss: 0.9932 - val_accuracy: 0.6761 - val_loss
: 0.9413
Epoch 18/100
1562/1562 ━━━━━━━━━━━━━━━━━━━━ 2s 1ms/step - accuracy: 0.6875 - loss: 0.8812 - val_accuracy: 0.6710 - val_loss:
0.9579
Epoch 19/100
1110/1562 ━━━━━━━━━━━━━━━━━━━━ 10s 24ms/step - accuracy: 0.6578 - loss: 0.9760
```

In [ ]: