EX.NO:14
27.10.25

IMPLEMENT A PRE-TRAINED CNN MODEL AS A FEATURE EXTRACTOR USING TRANSFER LEARNING MODEL

Aim

To implement a pre-trained CNN model as a feature extractor using transfer learning model.

Objective

To understand how a CNN pre-trained on ImageNet can extract visual features.

To freeze convolutional layers and use their output features for new tasks.

To train only the classifier head for faster and more accurate learning.

pseudocode

```
    Import Libraries
    Load a pre-trained CNN
    freeze all convolution layers to
prevent retrained
    Replace final classifier layer for
new number of classes.
     Load small dataset (CIFAR-10)
     Extract features
     Train only new classifier head
     Evaluate accuracy
```

~~objectives~~ observation

The pre-trained ResNet18 model was successfully loaded.

Frozen layers acted as fixed features extractor, capturing low-level edges, corners, textures.

output

```
epoch 1, LOSS : 0.8348
epoch 2, LOSS : 0.6229
epoch 3, LOSS : 0.5937
epoch 4, LOSS : 0.5797
epoch 5 ,LOSS : 0.5699
```

Training was significantly faster because only the final classifier parameter wire updated leading stable convergence.

This experiment demonstrated how transport learning can drastically reduce training and Computational cost.

Result

Successfully Implemented pre-trained CNN Model.

EXP NO: 15

27.10.25

IMPLEMENT A YOLO MODEL TO DETECT OBJECTS

AIM

To implement a YOLO model to detect objects

Objective

To study how YOLO detects multiple objects in a single image.

To understand the architecture and working a pre-trained YOLO model.

To use transfer learning for custom object decisions.

pseudocode

```
Install and import (YOLO package)
Load a pre-trained YOLO model
Load a test image or use a camera frame.
Run the model predict() method to detect objects.
Display bonding boxes and class label.
Save the annotated output image
```

Observation

The pre-trained YOLO V5S model successfully detected multiple objects such as cars, buses person in single frame.

image1 |1| content | IMG -20250523.WA001 jp

3 persons . 7.3 ms 640 X 480

Speed : 3 preporocess : 7.3ms inference 2.5 ms

postprocess pon image A shape

(1, 3, 640, 480)

output

Detected objects

Person

Person

person

Each object was enclosed in a bounding box with. a class label and Confidence score.

The inference time per image was very Low

The model demonstrated strong generalization without additional training.

The visualization clearly showed YOLO's ability to detect overlapping objects Complex scencs.

Result

Successfully Implement pre-trained YOLO.

RA2311047010011

10.1.38.19

Index

மரம் வளர்ப்போம்
மண் வளம் காப்போம்

Superior®
Note Books

Name : L DHARSHINI

Subject : DEEP LEARNING TECHNIQUES

SEC : .... Sec. : A Roll No. 011

School : ....