NAME: Renu Tamsekar

ROLL NO: 3253

TITLE : Image classification using CNN

LAB NO:

PROBLEM STATEMENT :  develop an algorithm and write a program for image classification using convolutional neural network.

METHODOLOGY :

* Import required libraries.
* Import the required csv file
* Split the data of the csv file into training and testing data.
* Convert the training data into float32 datatype since that is the required input format for CNN.
* Normalize the data.
* Import the sequential keras model.
* To this sequential model, add the basic CNN layers i.e. input layer, convolution layer, pooling layer, fully connected layer along with ReLu activation.
* Compile this model using SGD (stochastic gradient descent) optimizer and MSE (mean squared error) loss.
* Train the data over 100 epochs.

APPLICATIONS:

* Facial recognition.
* Medical imaging.
* Document analysis.
* Autonomous driving.
* Biometric authentication.

RESULTS :

* The loss is at its lowest (0.16) at the last epoch.
* Therefore the accuracy is at its highest (0.94) at the last epoch.
* The mode.summary() function shows the convolutional neural network created and how many trainable parameters each layer has and the output shape.

OBSERVATION :

* The loss goes on decreasing and the accuracy goes on increasing as the number of epochs is increasing.
* This indicates that the model is reaching its global minima in terms of error at every iteration.
* The model will show more accuracy when optimizer ‘adam’ is used.
* Cross entropy loss and mean squared error loss have different effects on the learning rate and convergence of the neural network.
* Cross entropy tends to have a faster convergence than mean squared error loss because it has a steeper gradient.

CONCLUSION :

* CNN proves to be better than other classifiers with accuracy of 94% in case of digital recognition.
* It doesn’t require human supervision for the task of identifying important features and therefore are very popularly used in image recognition and classification.
* Results can be made more accurate with optimizers and more number of hidden layers.
* For images typically 2D convolution is used, meaning that the kernel will be in a matrix form instead of a 1D convolution in which kernel will be a single number.

PRINT OF CODE AND OUTPUT