Project Report

Objective

The primary objective of this project was to develop a Convolutional Neural Network (CNN) model for image classification. The model was trained and tested on a dataset of grayscale images.

Data Preparation

The data used in this project was stored in CSV files. The `pandas` library was used to read the data and convert it into DataFrame format for easier manipulation. The data was then reshaped into a 4D tensor to be compatible with the CNN model.

Model Architecture

The model architecture consisted of several layers including Conv2D, BatchNormalization, MaxPool2D, Dropout, Flatten, and Dense layers. The model was compiled with the Adam optimizer, categorical crossentropy as the loss function, and accuracy as the evaluation metric.

Data Augmentation

An instance of `ImageDataGenerator` was created for data augmentation. This included featurewise centering and normalization, samplewise centering and normalization, rotation, zooming, width and height shifting, and validation split.

Model Training

The model was trained using the `fit` method of the `ImageDataGenerator` instance. The number of steps per epoch for both training and validation was calculated by dividing the number of samples by the batch size. The model was trained for 30 epochs.

Model Evaluation

The model's performance was evaluated using a confusion matrix. This provided a clear visualization of the model's performance on classifying images into their correct categories.

And then it has been observed that all the labels in the sample are 0. Therefore, it is not feasible to evaluate our model using this test data.

Conclusion

This project demonstrated the process of building, training, and evaluating a CNN model for image classification. It also highlighted the importance of proper data preparation in machine learning projects.