Problem:

Image classification by extracting features from image and applying knn classifier on the test data to predict results.

Solution:

1.Feature Extraction:

For Feature Extraction I have used lbpFeatures as it seemed more feasible compared to other functions. So in lbpFeatures we gave cellsize and Interpolation as my parameters. Now go by num of neighbors to access histograms cell by cell then normalize using L1 normalization and then reshape it into 1*N vector. This is basically bag of words implementation. Ideally I tried with different numofneighbors then I narrowed it down to 4 as accuracy is way better.

2. Model fitting and training:

After we extract features for all images in train and test. We use **fitcknn** to train the model and the we use predict to classify the test data. If we use too many numofneighbors it reduces the accuracy as they will be so many randoms on the same cluster. If we use less there might not be enough data to classify. So used optimal neighbors of 10 in fitcknn.

The main thing in this as we have very less data to train and too many categories to predict. It's much harder to classify test data. Tried different stuff using kmeans clustering and then create new frequency vector and classify but it only got worse.