Principles of Biological Vision

Assignment2: Landscape Recognition with Bayesian Classifier

Objectives:-

This assignment is related to holistic features for context recognition discussed in module 04-03 in the class. We are required to use the landscape recognition image dataset from Kaggle, extract the GIST features for the images as described in the reference (Oliva & Torralba – see slide 14), and develop a Bayesian classifier for the different categories of the landscapes (e.g., coast, desert, forest, etc.) provided in the dataset. We shall also need to evaluate the performance of your classifier and create a report.The project involves

- [ ] **Objective 1:** Pre-processing of the images and extraction of GIST features.
- [ ] **Objective 2:** Programming classifier in the form of Bayesian network with suitable configuration for the stated purpose.
- [ ] **Objective 3:** Estimating the parameters of the network (priors and conditionals), as discussed in module 03-05.
- [ ] **Objective 4:** Testing the classifier and performance evaluation.

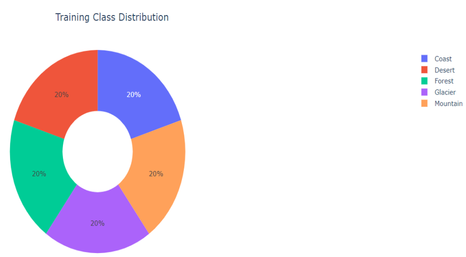
Major Python libraries and its dependencies :

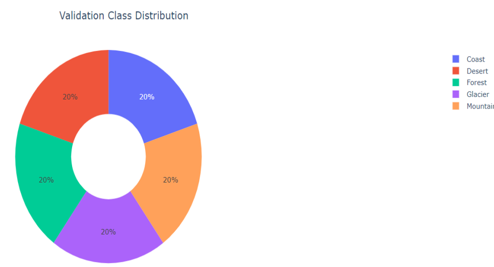
Libraries	**Functions**
CV2	Operations on images.
Numpy	Matrix Math etc.
Matplotlib	3D Graphs plots etc.
Scikit learn	Evaluation and Interpretation

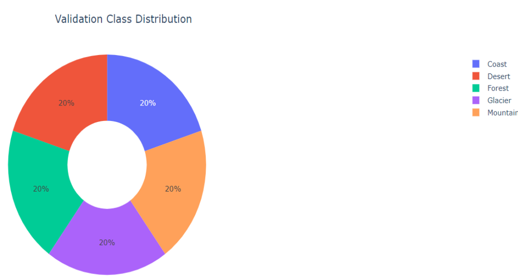
Objective 1: Data Exploration and Preprocessing

Approach/Implementation Explanation:

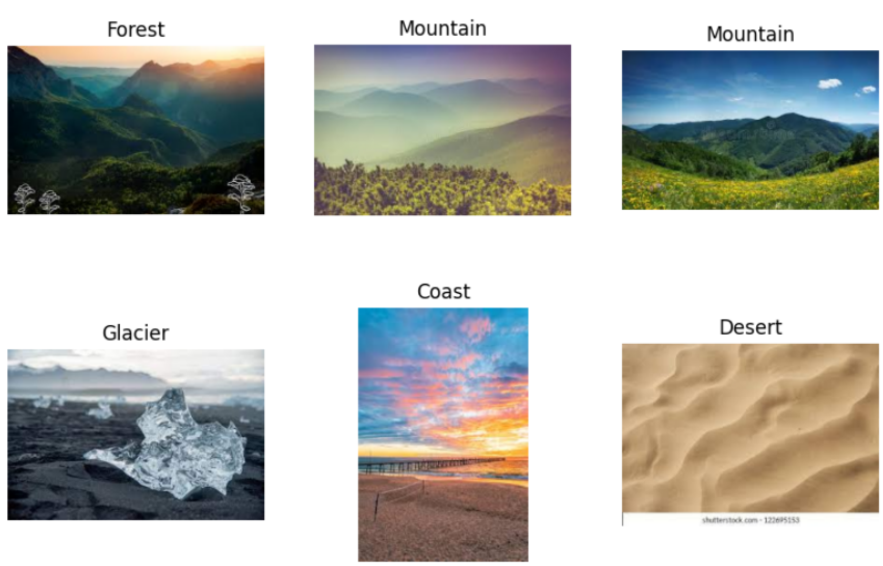
- The dataset used is from kaggle. It is loaded using kaggle api. The link for the same is : [Click here](#) .
- Further exploration is done with the training , testing and validation dataset. Each dataset consists of 5 classes with an equal number of datapoints. Following are the pie charts for the same:



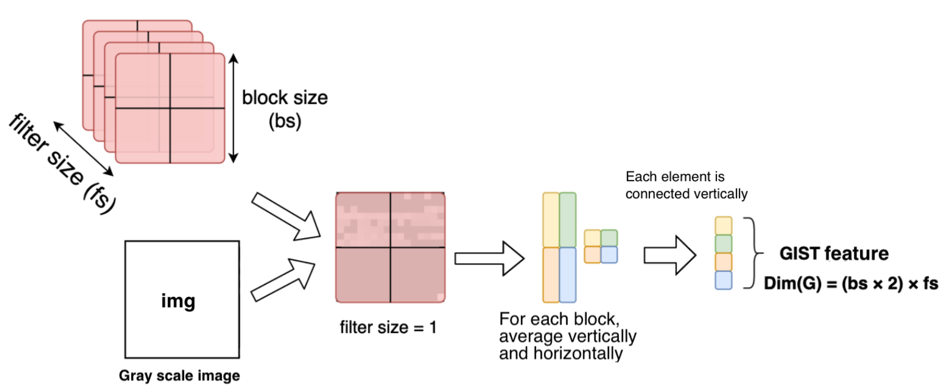




- Visualization of some images are done using cv2 . Following are few images with corresponding labels:



- Now GIST features are extracted. To do the same following steps can be taken :
  - Image Preprocessing: Preprocessing is a technique used to improve the consistency and quality of an input picture before extracting Gist features, such as resizing, grayscale, and histogram equalization.
  - Partitioning Images: The picture is split up into a collection of non-overlapping chunks or areas using gist feature extraction. This division process facilitates content analysis of the picture at various spatial scales.
  - Refinery Bank: Filters like Gaussian, Gabor, and simple gradient filters are applied to each block or section of an image, capturing various visual information like texture, edges, and gradients due to their orientation and multi-scale nature.
  - Feature Dictionary:By statistically measuring the filter responses, features are calculated for every block and filter combination. The mean, standard deviation, skewness, and kurtosis are examples of common statistics. The texture and edge information found in each block is summed up in these statistics.
  - Mixing: Gist representation is created by combining block characteristics and filters using max or mean pooling techniques, with the final representation varying based on the chosen pooling mechanism.
  - Combined or Histogram: A value histogram is constructed by concatenating the pooled features from each block and filter. After completing this stage, the global information of the whole picture is encoded in a one-dimensional feature vector.
  - Standardization: It is usual practice to apply feature vector normalization in order to make the Gist feature resistant to fluctuations in lighting, contrast, and size. This can use methods such as Z-score standardization or L2 normalization.
  - Last Principal Feature:The final Gist representation of the image is the normalized feature vector. For applications such as image retrieval, scene identification, and picture classification, this vector may be fed into a variety of machine learning techniques.



Objective 2:Bayesian Network Classifier

Approach/Implementation:

- The Bayesian network classifier is a probabilistic machine learning algorithm that is based on Bayes' theorem.The algorithm calculates the probability of a data point belonging to a particular class based on the probabilities of its individual features. The class with the highest probability is the predicted class.
- The algorithm calculates the probability of a data point belonging to a particular class based on the probabilities of its individual features. The class with the highest probability is the predicted class.
- It calculates class probabilities (P(class)) and feature probabilities (P(feature | class)) for each feature-value combination within each class.
- The fit() method is used for training the Bayesian network classifier. It takes two input arguments, X (features) and y (labels). It calculates class priors and feature likelihoods for each feature and class combination.
- Following are the steps:
    - Calculate the unique classes and their counts in the training data.
    - Calculate class priors by dividing the count of each class by the total number of samples.
    - For each feature, calculate feature likelihoods for each class and feature value.
    - Feature likelihoods are calculated as the conditional probability of a feature value given a class.
    - The results are stored in the self.class_prior and self.feature_likelihoods dictionaries.
- The predict() method is used for making predictions on new data.
- It receives an input sample matrix, X, and outputs a list of expected class labels for every sample.
- Based on the class prior and feature likelihoods, it computes the posterior probability of each class for each sample.
- The anticipated class label is determined by selecting the class with the highest posterior probability.
- For additional examination or analysis, the posterior probabilities for every class are kept in the self.posterior_probability list.

Objective 3:Training the Bayesian network - Parameter estimation

Approach/Implementation:

- The training data is trained using the above bayesian network classifier.
- Estimated the parameters of the Bayesian Network using the training data.
- Calculated the prior probabilities (P(C)) for each landscape category.
- Estimated the likelihood of observing a specific GIST feature value given a landscape category (P(X|C)).
- Now values are predicted on testing data and an accuracy of 53.8 percent is achieved.
- This model could not achieve 90 percent accuracy because in this model features are assumed independent of each other which may lead to poor training and hence decreasing the accuracy on testing data. The above model is a simplified version of Bayesian Network Classifier. If we can implement a network which can use complex probabilistic graphical models for probability calculations we could have achieved a better accuracy. Furthermore if we use a neural network we can get 90 percent accuracy.

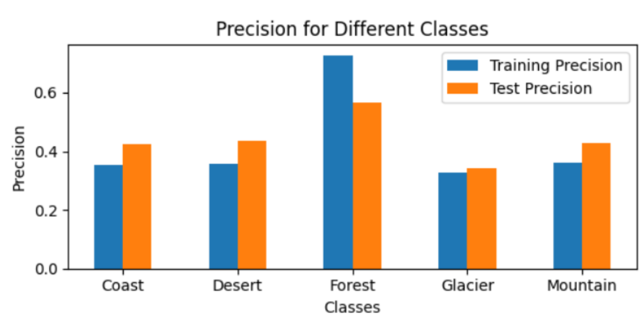
Objective 4:Evaluation and Interpretation

Approach/Implementation:

- For evaluation and further interpretation precision, recall, and F1-score for each individual classes are calculated.
    - Precision: Precision is a widely used performance parameter in machine learning and statistics, particularly in classification tasks. It is one of the primary assessment metrics used to gauge a classification model's quality since it quantifies the model's accuracy of positive predictions. The following formula is used to determine precision:

      Precision= $True\ Positives/(True\ Positives\ +\ False\ Positives)$

      Below is the bar graph representing the precision score for different classes on training data and testing data :

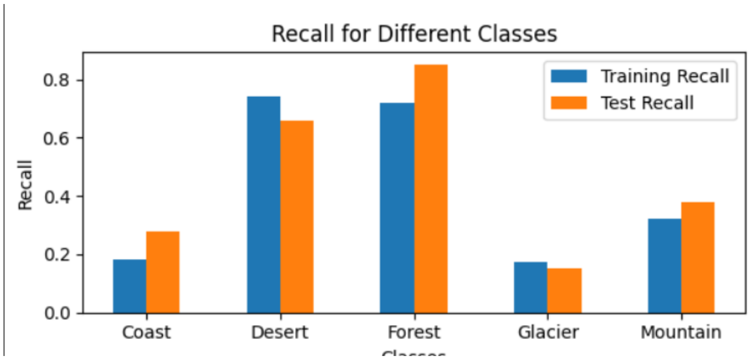
      

      Precision is a measure of the model's ability to make accurate positive predictions.

- Recall : Information retrieval, data analysis, and machine learning all employ recall as a performance indicator. Specifically, it evaluates the ratio of true positives to the total number of real positive occurrences, indicating how well a model or system can detect all relevant instances within a dataset. It goes by the names hit rate, sensitivity, and true positive rate as well.

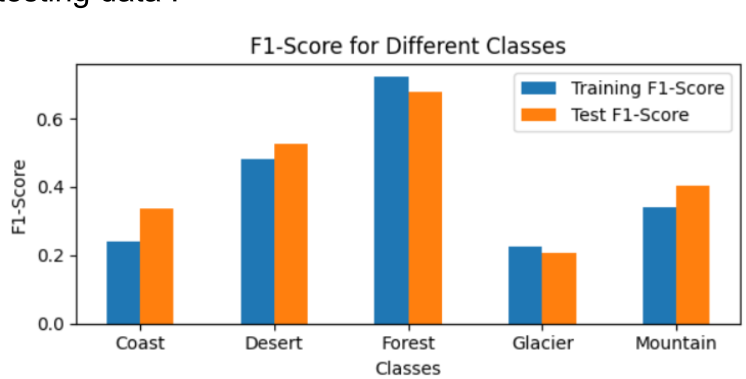
  Precision= $True\ Positives/(True\ Positives\ +\ False\ Negative)$

  Below is the bar graph representing the recall score for different classes on training data and testing data :

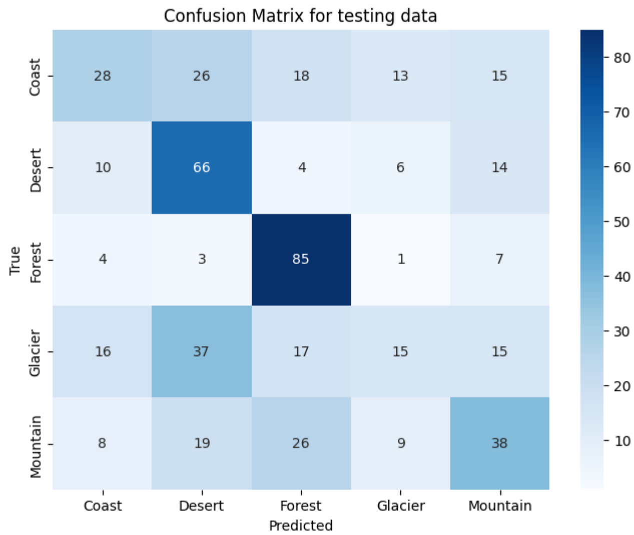
  

  Recall is particularly important when the cost of missing positive instances is high.

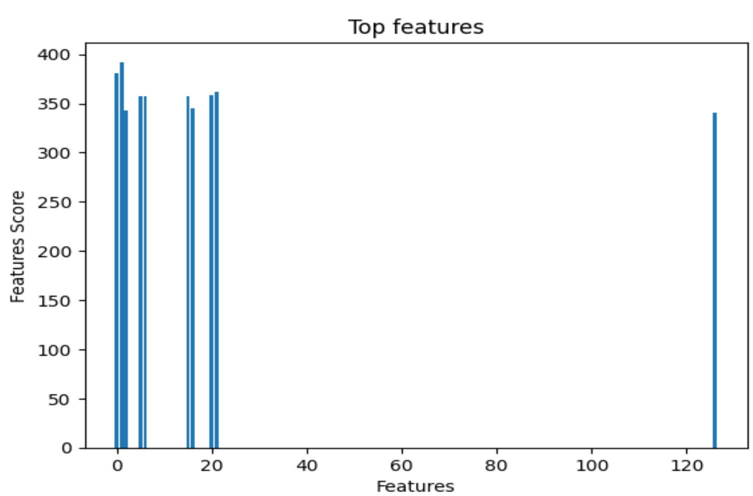
- f1 score : The F1 score is a commonly used metric in machine learning and statistics, particularly in classification tasks. It is a single numerical value that summarizes the performance of a classification model, taking into account both precision and recall. The F1 score is computed as the harmonic mean of precision and recall. This means that it penalizes extreme values more severely than the arithmetic mean, making it especially useful when precision and recall have significant differences. When either precision or recall is very low, the F1 score will be much lower than the arithmetic mean, highlighting the imbalanced performance. Below is the bar graph representing the f1 score for different classes on training data and testing data :

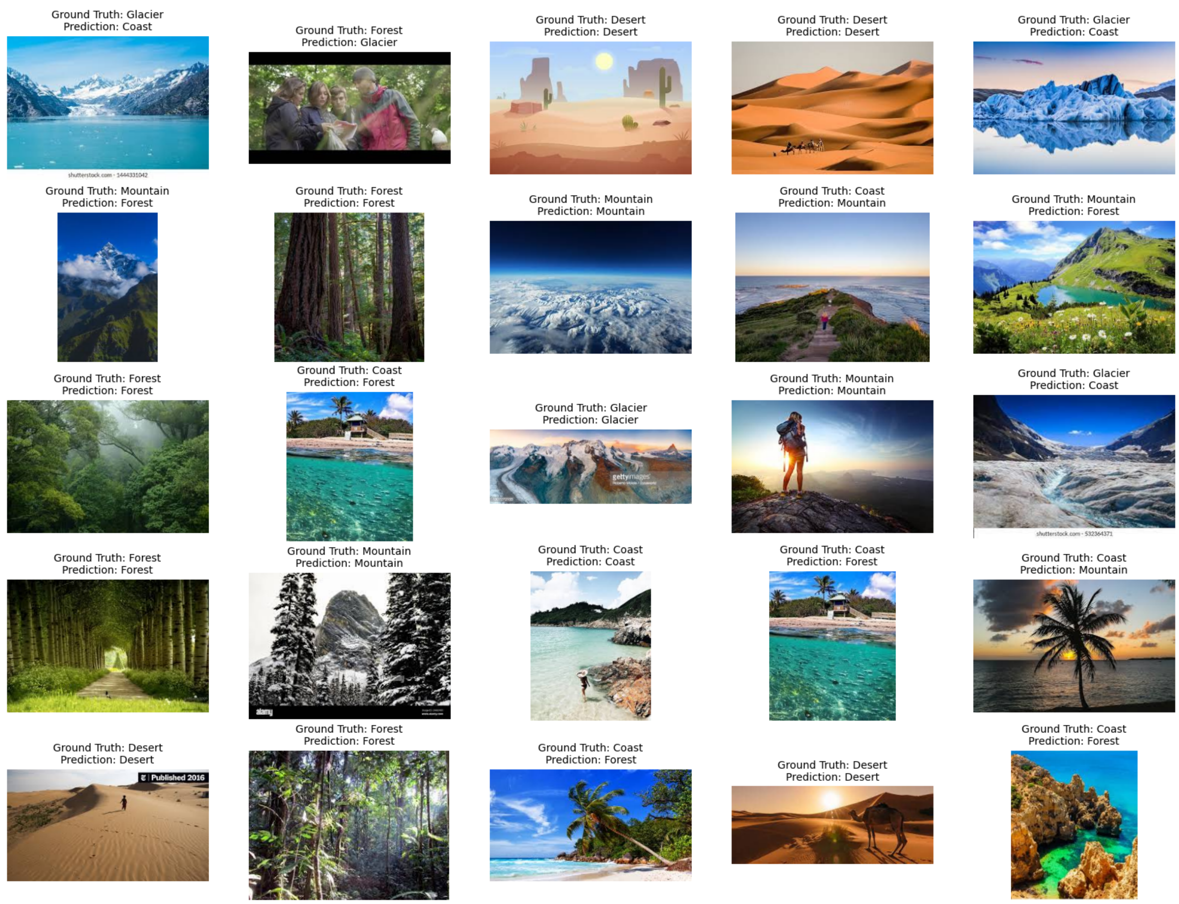
- Confusion matrix for testing data is also visualized as a heat map. Following is the result:



- From above graphs and heat map we can say that the forest class is the class with the most true positives then desert then mountain then coast and then glacier. The possible reason for this may be that there could be characteristics that set the "forest" class apart from the others, making it very recognisable and unique. These elements could consist of trees, verdant vegetation, and particular textures that are typically found in scenes set in forests. The glacier seems to have the least recognizable characteristics. Some of the glaciers might have been recognized as mountains due to similarities with huge ice blocks and some glaciers might have been recognized as coast due to presence of water. Similarly other classes could have been predicted to other classes due to similarities with those classes.
- After all the analysis from the metrics we got a feature importance analysis is also done to identify which specific features contribute the most to the classification decisions. For this from sklearn.feature_selection SelectKBest is used which selects the top k features from a dataset based on some scoring function. Below is the bar graph representing top 10 features with their scores:



- The feature with values 0 ,1 and 21 seems to be the 3 most significant features in predicting the target variable.These features seem to be related to image contrast, brightness, or intensity as variations in image contrast or lighting conditions strongly affect the prediction of image class. These three features are main features in classifying images into different types.
- Now a subset of testing data is chosen randomly and the model's predictions are visualized along with the ground truth labels.



- Also the highest and second highest posterior probabilities are calculated along with their corresponding classes. The highest probabilities are of those classes which are predicted in the above image. The posterior probabilities for the first five images are :

```
Predicted class is: Coast - with posterior probability 7.893253086385026e-40
Class with second highest posterior probability is: Mountain - with posterior probability 4.5181919193438033e-41

Predicted class is: Glacier - with posterior probability 1.35923112133526e-19
Class with second highest posterior probability is: Desert - with posterior probability 4.0854898383981506e-20

Predicted class is: Desert - with posterior probability 3.90791156072151 15e-09
Class with second highest posterior probability is: Glacier - with posterior probability 3.163792638426322e-10

Predicted class is: Desert - with posterior probability 8.30248870045118e-11
Class with second highest posterior probability is: Glacier - with posterior probability 1.0021054881986274e-11

Predicted class is: Coast - with posterior probability 2.5338073391681963e-35
Class with second highest posterior probability is: Mountain - with posterior probability 1.3984486728287903e-36
```

- Following are the instances where the model made correct or incorrect predictions :

<u>Correct predictions:</u>

The forests are correctly predicted due to the presence of trees, verdant vegetation, and particular textures that are typically found in scenes set in forests. All these are important characteristics that set a forest class apart from others. The glaciers and coast are predicted correctly to the presence of snow and ocean in the scene. There are no such conflicting features which could lead to incorrect classification of them.

<u>Incorrect predictions:</u>

The first image is predicted as a forest instead of a mountain due to the presence of a good amount of greenery which is a key characteristic of forest scenes. The second image is predicted as a coast instead of a glacier due to the presence of a good amount of water (inferred as ocean) which is a key characteristic of coasts.Similarly for the third image due to presence of more trees than water it is predicted as forest instead of a coast.

Conclusion summarizing the findings and lessons learned

From the assignment I learned how we can use Bayesian Classifiers for landscape recognition. I have learned about the basics of GIST features extraction and how these features can be used for classification. Then I learned about the Bayesian network Classifier and its working through a simple implementation and got to know about its pros and cons. I also learned how the network can be improved to improve the accuracy like we can use neural networks or a more complex Bayesian network .

References :

https://github.com/imoken1122/GIST-feature-extractor/blob/master/gist.py

https://www.kaggle.com/datasets/utkarshsaxenadn/landscape-recognition-image-dataset-12k-images