# Terrain Classification with Satellite Imagery

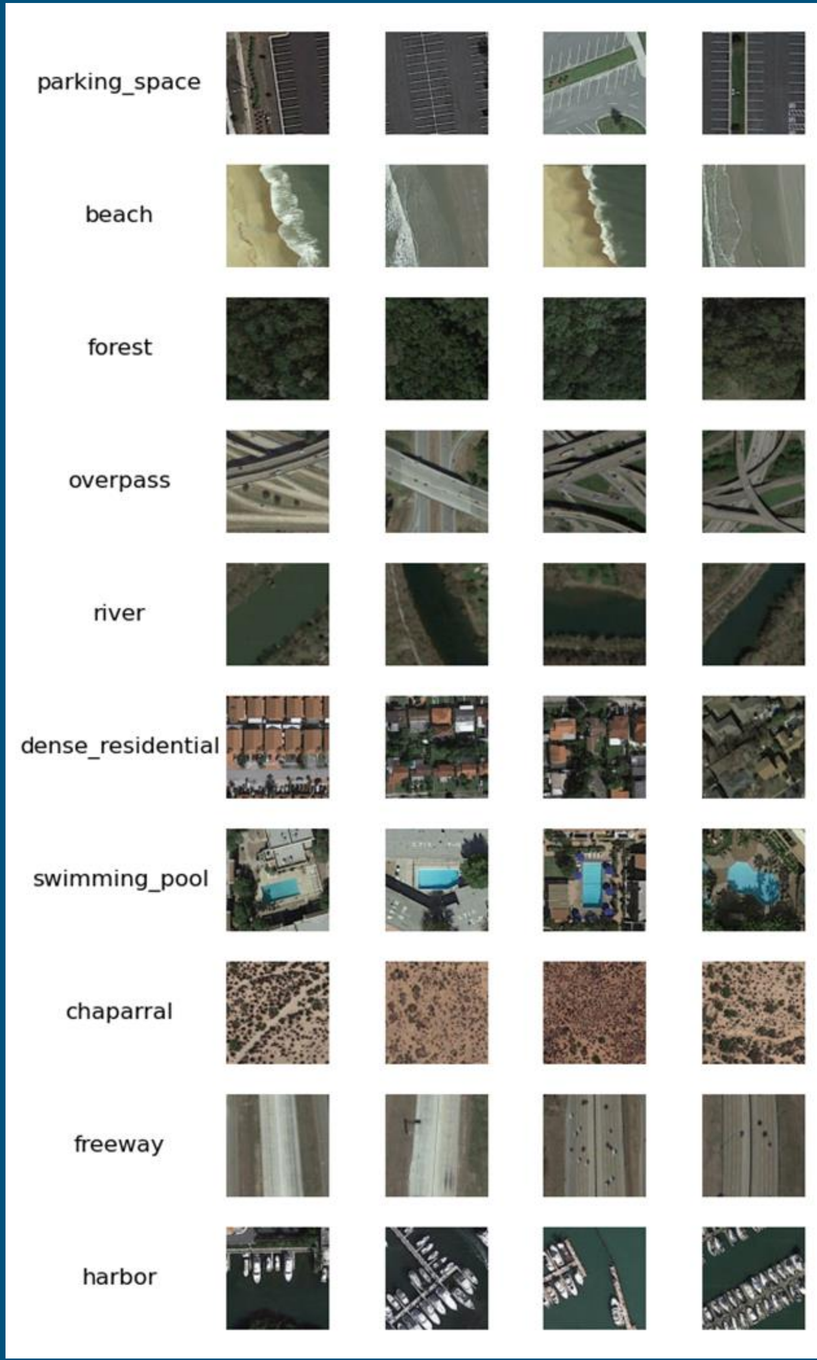
April 15th, 2024

MIDS W281 | Spring 2024 | Final Project

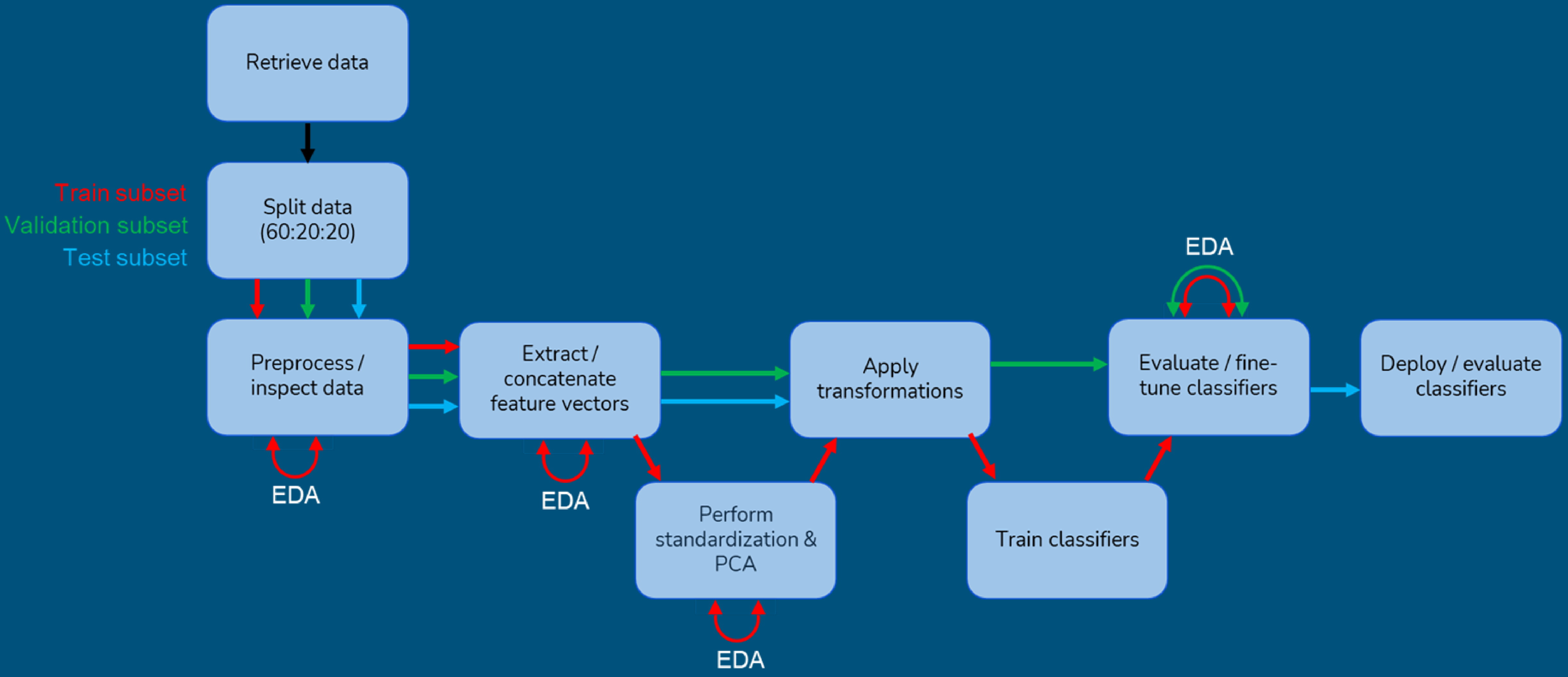
J. Spencer Morris, Muthumayan Madhayyan, Maiya Caldwell

# Image Dataset

- Motivation
  - Classification of satellite imagery scenes enables understanding in land use and changes in scenery
  - Applications in climate studies, defense and war, navigation and mapping, agricultural and urban planning, etc.
- Dataset
  - PatternNet is a high-resolution remote-sensing dataset generated using Google Earth imagery
  - 800 JPEG images for each of 38 classes
  - Images were manually labeled
  - 10 classes were chosen for inclusion in the analysis



# Approach



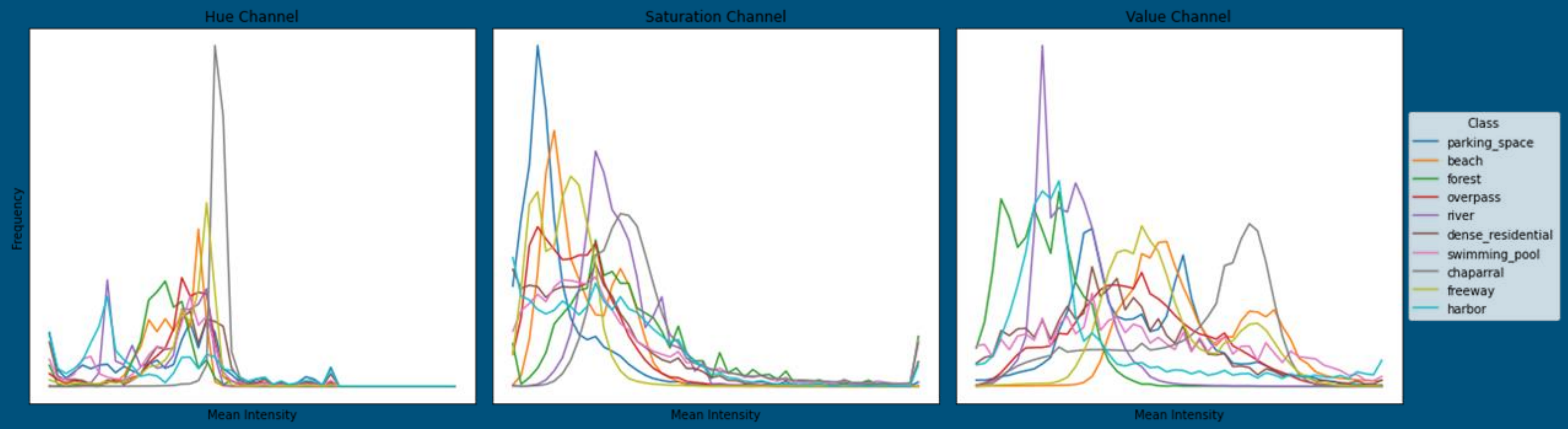
Retrieve data
Train subset
Validation subset
Test subset
Split data (60:20:20)
Preprocess / inspect data
EDA
Extract / concatenate feature vectors
EDA
Perform standardization & PCA
EDA
Apply transformations
Train classifiers
EDA
Evaluate / fine-tune classifiers
Deploy / evaluate classifiers

# Extracted Features

- Each image represented by 1412 features
- Mean and standard deviation of each red, green, and blue channel
- Histogram of HSV values
- Flattened HOG values
- Mean GLCM values
- Frequency spectrum histograms
- SIFT features with Bag of Visual Words applied

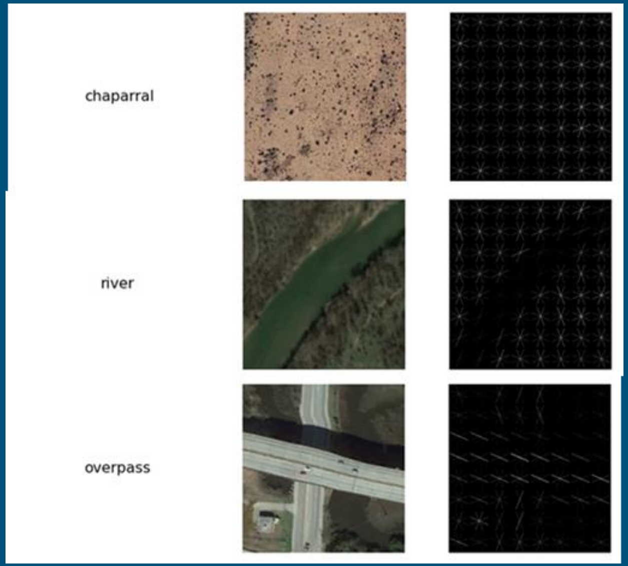
# Color Features

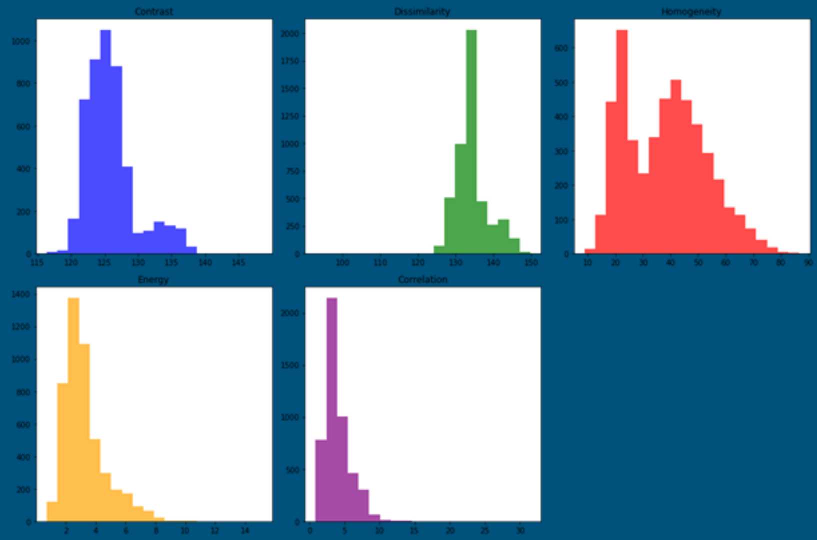
- RGB
  - Raw RGB color channel data were presumed useful during visual inspection of image classes
  - Simple statistics (mean and standard deviation) were computed for each color channel of an image
  - Each image had 6 RGB features
- HSV
  - Hue, saturation, and value distributions were presumed useful when computing mean histograms for each class
  - Histograms for each image were consistently binned to determine an approximation to the HSV distribution
  - Each image had 75 HSV features



# Orientation and Texture Features

- Histogram of Orientation Gradients (HOG)
  - The distribution of gradient orientations is a useful descriptor for local shape and edge information
  - Images were converted to grayscale and a 32x32 patch was used to compute HOG features
  - Each image had 1296 HOG features
- Gray-level co-occurrence matrices (GLCM)
  - Textural qualities are determined by measuring the frequency of co-occurring pixel intensities
  - Images were converted to grayscale then the mean of each GLCM metric was computed given a fixed pixel-offset at 4 angles
  - Each image had 5 GLCM features



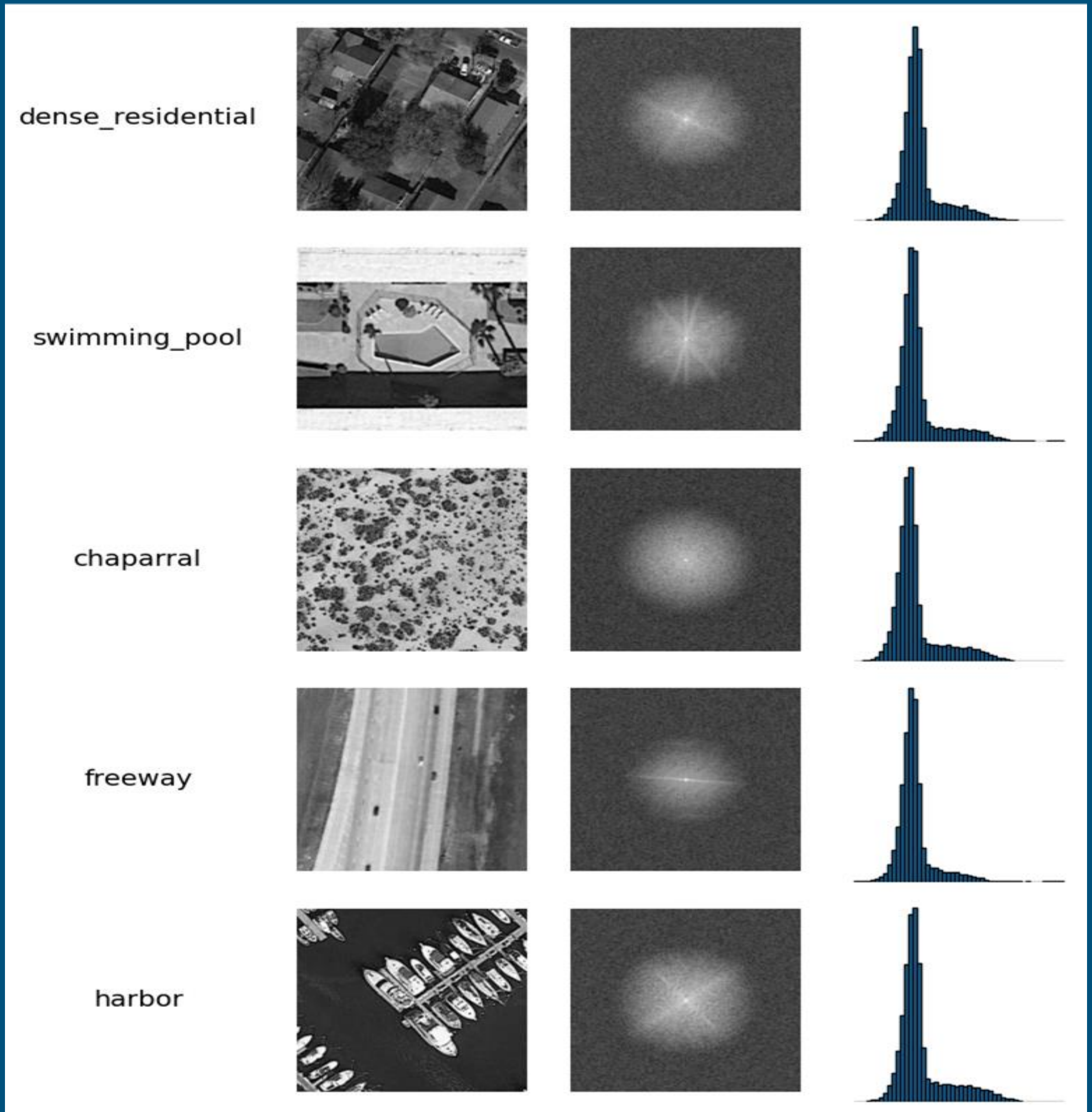



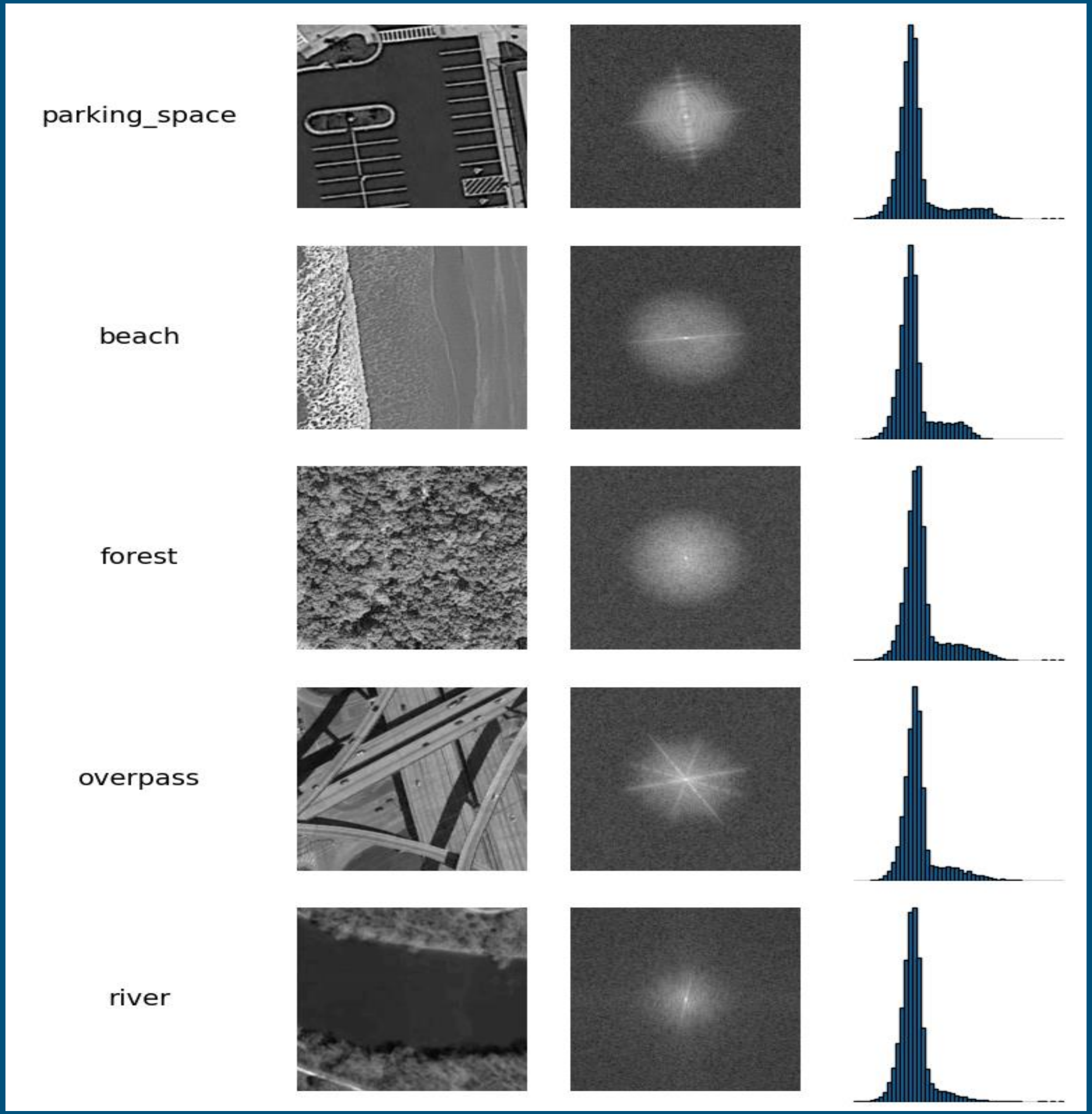
# Spatial Frequency Distribution

- Distribution of spatial frequencies vary between images of different classes
- Feature extraction in frequency domain is relatively time-efficient; for about 4500 images, extraction takes < 25 seconds
- The distribution of spatial frequencies can be leveraged as **feature vectors** in image classification
- We ‘bin’ the frequency distribution into 25 bins
- Observations:
  - PCA on this feature vector yields < 10 components
  - Training time for this feature vectors is small (< 70 seconds)
  - Accuracy is between 85 - 87 % on this validation dataset

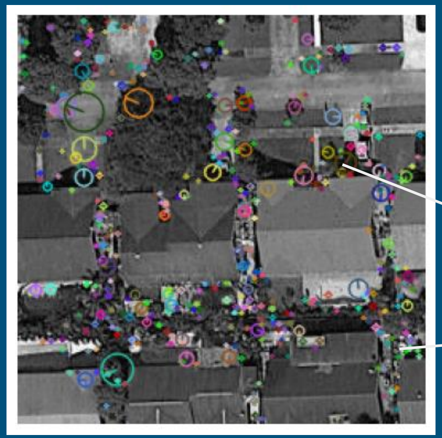
# Spatial Frequency Spectrum

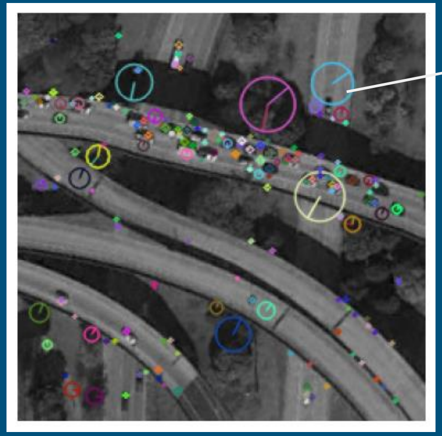
Frequency Histograms



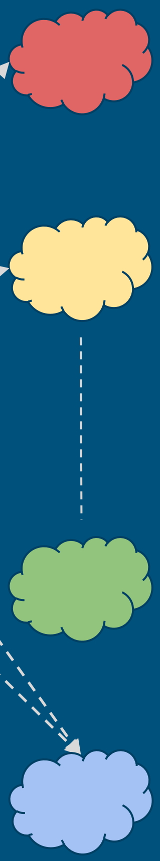



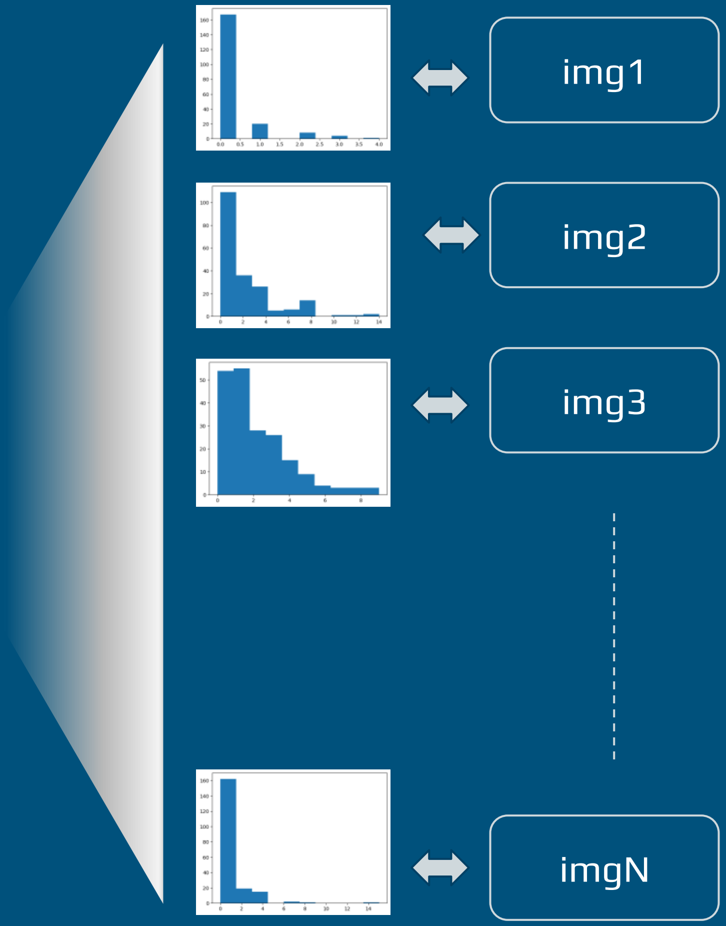
# SIFT Features





| | Class | D1 | ... | D128 |
|---|---|---|---|---|
| KP1 | | | | |
| KP2 | | | | |
| ... | | | | |
| ... | | | | |
| KPn | | | | |






Keypoint extraction

Edge & contrast thresholds to constrain # of keypoints

Bag of Visual Words clusters using K-Means

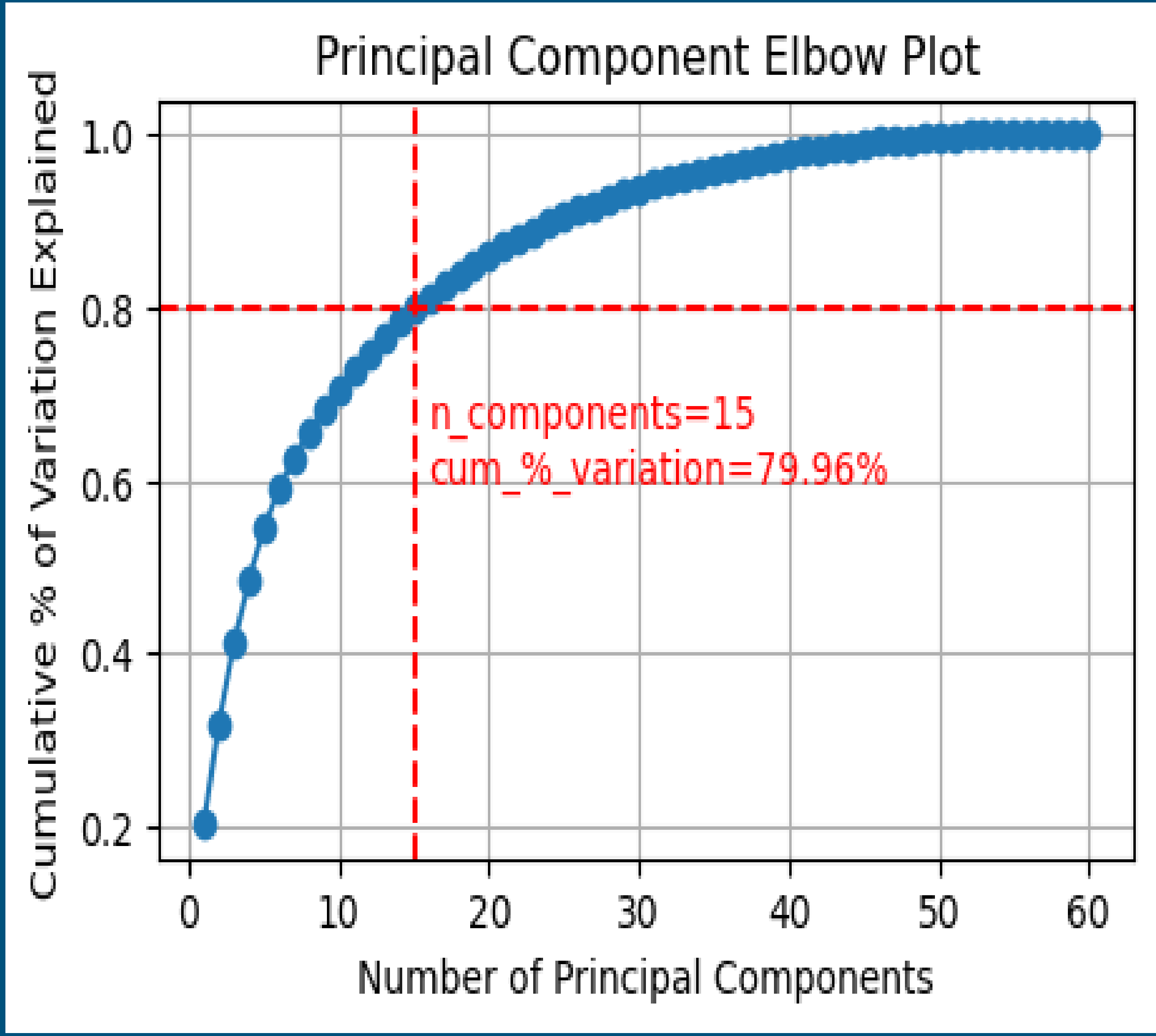
Tunable number of clusters

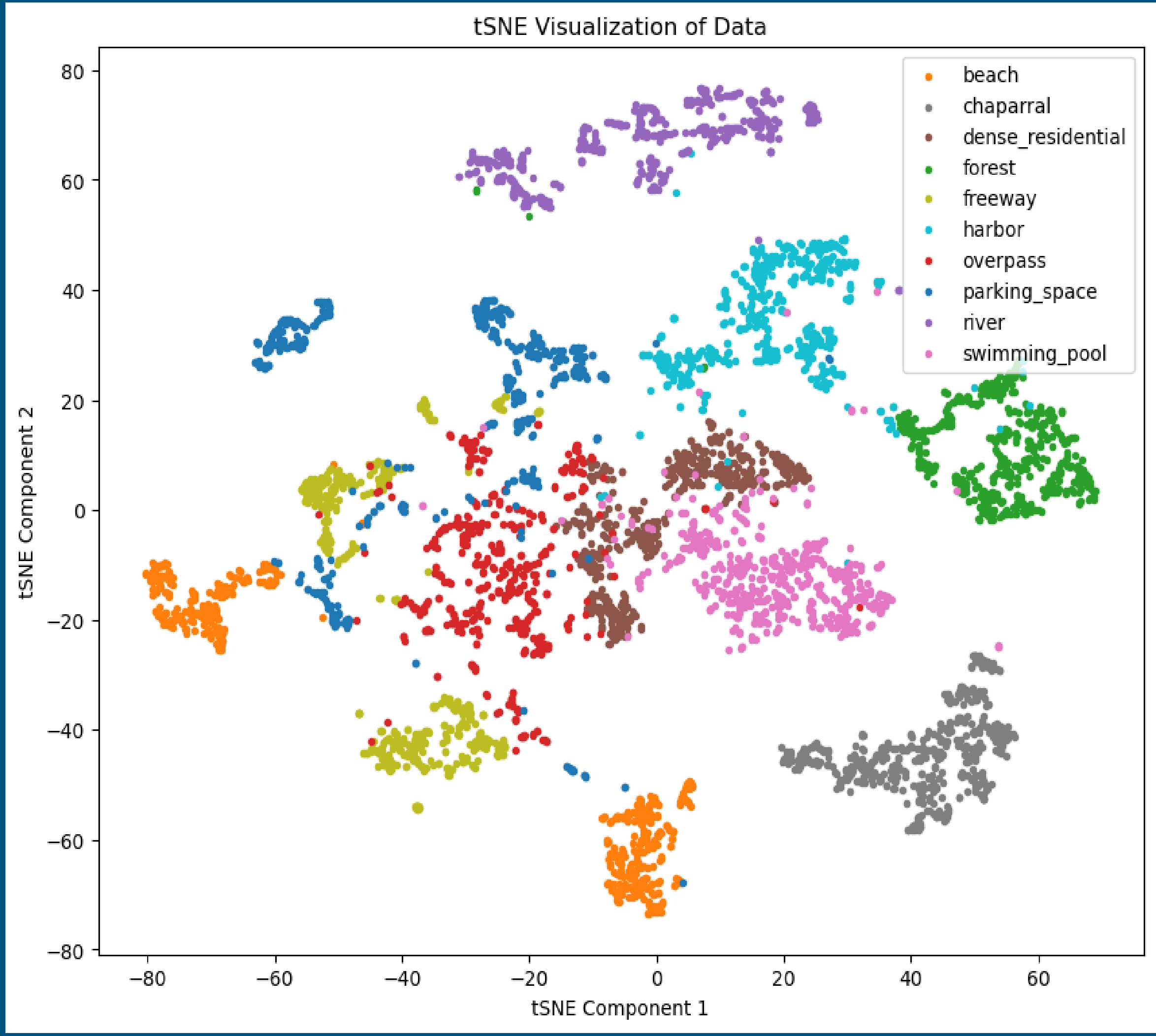
Distribution of words used as feature vectors

# PCA and tSNE

- Some of the feature vectors are high dimensional
  - Examples: SIFT, HOG
- Higher dimensional features incur more computational cost; PCA provides a simpler way to reduce dimensionality.
- Run PCA to identify features that have maximal impact to variability in the dataset.
- Run tSNE to visualize clustering of data based on the reduced feature vector set identified in the PCA step.
- Subsequent stages will subject the data to standard scaling and then do a PCA fit-transform to work on the more important feature vectors.

# PCA, tSNE with HSV vectors



# Classifiers

## Random Forest Classifier

- Ensemble classifier based on decision trees
- Suited for multiclass problems
- Scales with large datasets
- Hyperparameters: number of trees, number of features, max_depth determined by grid search

## Support Vector Machine

- Uses one-against-rest approach for multiclass problems
- Large dimensions can impact the training efficiency
- Relies on distances separating class boundaries
- Can work with linear, polynomial and radial boundaries; this makes it really useful for some type of data

# Model Optimization

- GridSearch
- All features:
  - Random Forest
    - Estimators: 100, 200, **300**
    - Max depth: none, 10, **20**
  - SVM
    - C: 0.1, 1, **10**
    - Kernel: linear, **rbf**
- HSV and GLCM only (best performing feature types):
  - Random Forest
    - Estimators: **100**, 200, 300
    - Max depth: **none**, 10, 20
  - SVM
    - C: 0.1, 1, **10**
    - Kernel: linear, **rbf**

# Accuracy vs Efficiency

| Feature | Extraction Time | # Components & Cumulative % Variance at Inflection Point* | Training Time | | Accuracy | | |
|---|---|---|---|---|---|---|---|
| | | | RFC | SVM | RFC | SVM | |
| RGB | 0:00:11.83 | 2 (97.2%) | 0:00:43.06 | 0:00:10.78 | 0.6823 | 0.677 | |
| HSV | 0:00:28.32 | 15 (80 %) | 0:01:21.79 | 0:00:07.59 | 0.9681 | 0.9843 | 1 |
| HOG | 0:00:37.142 | 165 (94.60%) | 0:05:57.54 | 0:10:38.53 | 0.86241 | 0.91994 | |
| GLCM | 0:02:20.250 | 3 (97.6 %) | 0:00:40.11 | 0:00:06.93 | 0.88555 | 0.8824 | 2 |
| Spatial Freq | 0:00:42.996 | 7 (90.51%) | 0:01:07.16 | 0:00:09.66 | 0.8549 | 0.8749 | |
| SIFT | 0:25:43.613 | 5 (92.4%) | 0:01:06.80 | 0:00:08.55 | 0.320 | 0.359 | |
| All the above | 0:29:32.39 | 182 (93.8%) | 0:05:48.09 | 0:00:24.03 | 0.9349 | 0.9881 | |
| HSV + GLCM | 0:02:50.67 | 17 (82.3%) | 0:01:39.01 | 0:00:06.57 | 0.9781 | 0.9912 | |

- Based on 4500 training images across 10 classes
- Accuracy based on 1600 validation images across 10 classes
- *Number of PCA components with highest explained variance

# Test Evaluation

- Our dataset contains 1600 images across the subset of 10 classes; we had isolated about 20% of images at the start of the project for testing
- We chose the the two best feature types that have the best combination of efficiency and accuracy: GLCM and HSV
- Using the test set, we evaluated the Random Forest and SVM classifiers with the chosen feature types

Random Forest Classifier

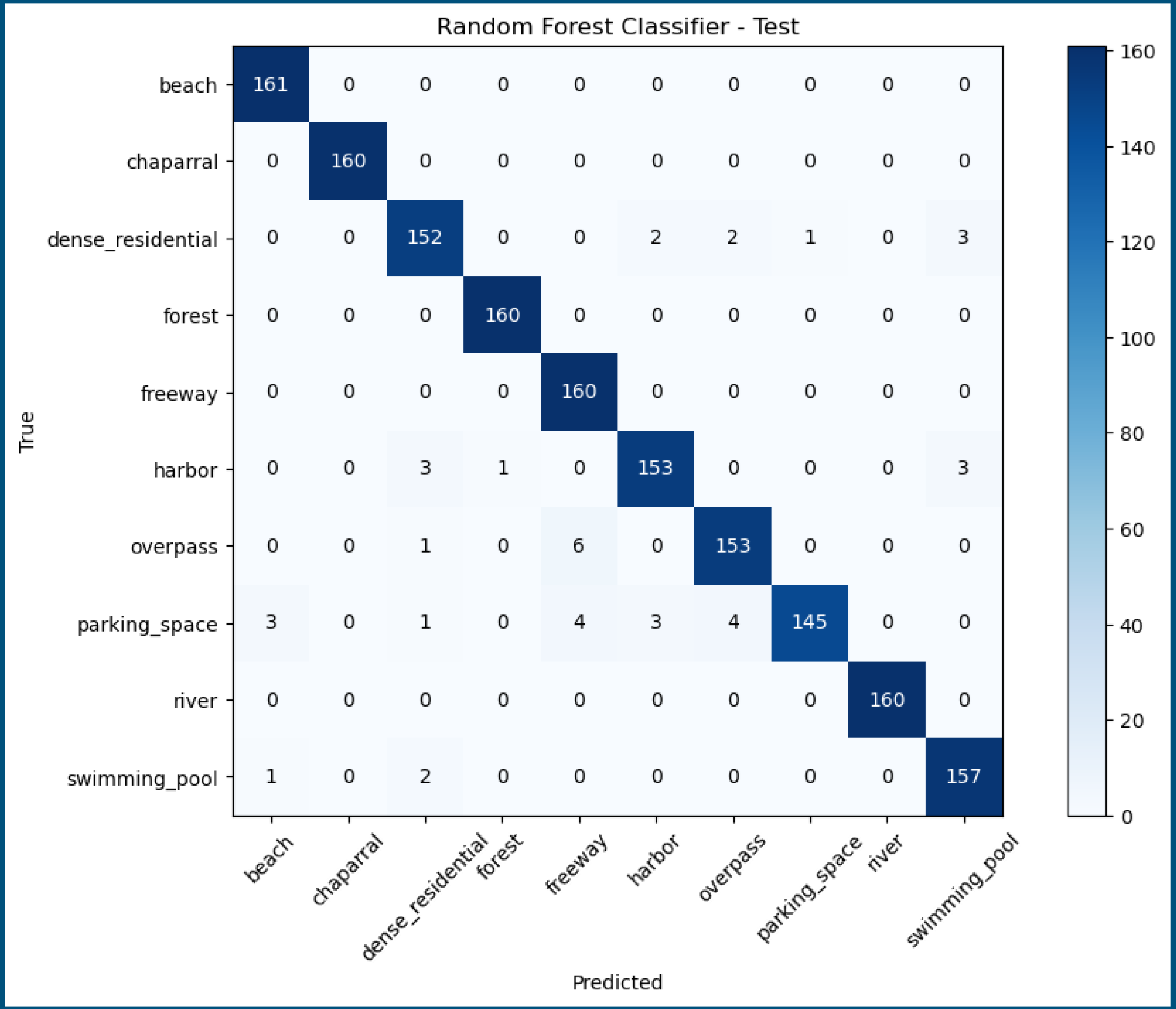
Train Accuracy: 1.0
Test Accuracy: 0.9781113195747342

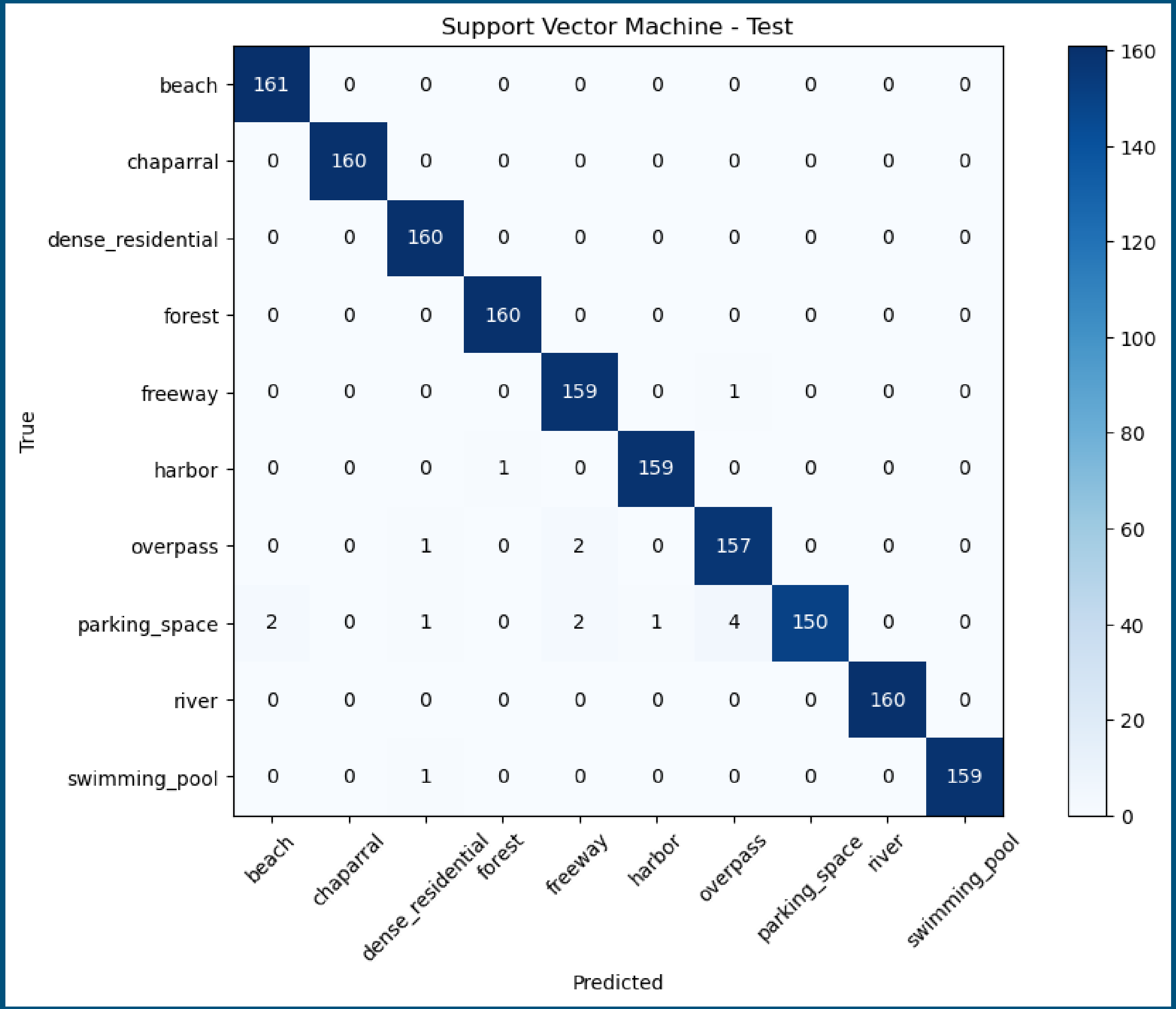
Support Vector Machine

Train Accuracy: 0.9972911023129819
Test Accuracy: 0.9912445278298937

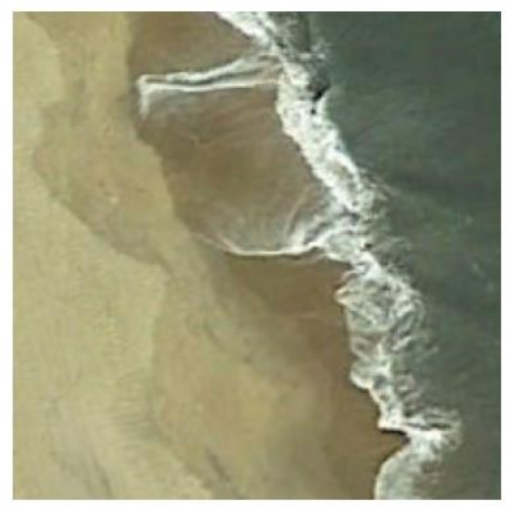
# Test Evaluation

HSV + GLCM Features



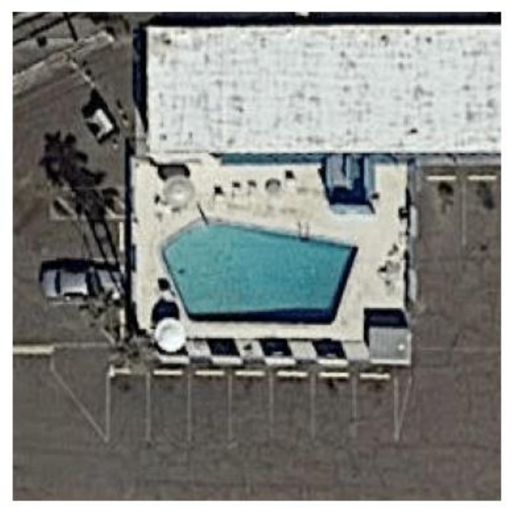



# Example Misclassifications



True Class: beach
Predicted Class: parking_space

True Class: Harbor
Predicted Class: parking_space



True Class: swimming_pool
Predicted Class: dense_residential

True Class: parking_space
Predicted Class: overpass

# Summary

- Terrain surveillance image dataset
- EDA
- Extracted various features and aggregated into feature vectors
- Most of the components that explained the highest amount of variance using the full concatenated feature vector were HOG
    - More than 80% of the 182 resulting components were HOG features!
- Found using HSV features alone in our classifiers surprisingly performed very well
- HSV + GLCM performed optimally
- SVM

# Thank you!

# Q&A