**README**

* A pre-created vocabulary from the previous “Homework 6” is used in this program.
* Run the program.
* Enter “1” to analyze training samples, “2” to analyze test samples and “3” to analyze the sixth class “bookshelf” samples.
* Check the results in “results.txt”
* “train\_results.txt”, “test\_results.txt” and “bookshelf\_results.txt” are provided with program to show precalculated results of the 3 running option.

**EE 576 HW7**

**Methodology:**

Same vocabulary is used as the previous homework. Labeled the bow histograms of the images and with these data and trained OpenCV SVM. One vs All strategy is chosen for this assignment. The reason mainly is that I am using OpenCV SVM. In OpenCV SVM, if you try to implement one vs one strategy via multiclassing. It creates voting scheme inside the API and thus we cannot intervene before the voting scheme to threshold the distances. In one vs all, I trained 5 different classifiers ( 1 for each class) and chose positive samples as the respective class for labeling and took the distances as output. Thresholded them with 3 different threshold and evaluated the results.

**Results for Train Data**

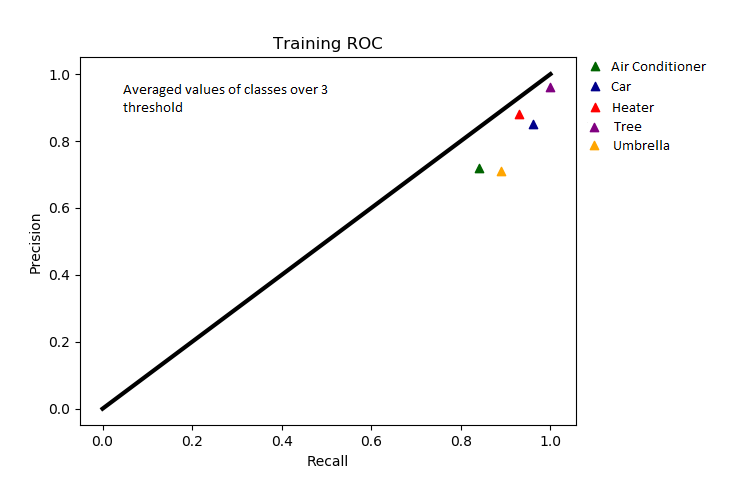
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Threshold = 0** | **Air Conditioner** | **Car** | **Heater** | **Tree** | **Umbrella** |
| **Precision** | 0.85 | 0.97 | 1 | 1 | 0.75 |
| **Recall** | 0.73 | 0.93 | 0.87 | 1 | 0.8 |

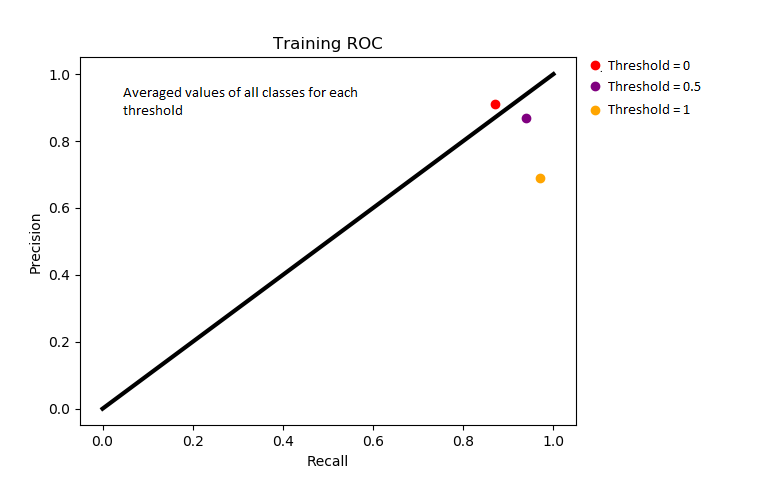
*Every column represents the classifier which trained with positive samples of the respective class name.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Threshold = 0.5** | **Air Conditioner** | **Car** | **Heater** | **Tree** | **Umbrella** |
| **Precision** | 0.76 | 0.88 | 0.94 | 1 | 0.75 |
| **Recall** | 0.86 | 0.96 | 0.96 | 1 | 0.9 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Threshold = 1** | **Air Conditioner** | **Car** | **Heater** | **Tree** | **Umbrella** |
| **Precision** | 0.56 | 0.7 | 0.69 | 0.88 | 0.63 |
| **Recall** | 0.93 | 0.97 | 0.97 | 1 | 0.97 |

As we see from the threshold ROC graph, increased threshold is a trade of between precision and recall. For larger thresholds, recall increases and precision decreases. This means, algorithm becomes less selective and finds most of the positives but at the same time its error rate increases. Threshold = 1 is a steep decrease in performance and threshold = 0 and threshold = 0.5 is comparable trade-offs. Considering the test results, we can say threshold = 0.5 is the optimum result for our algorithm. For classes best results belongs to the “Tree” class and the “Air conditioner” has the worst results. As we remember from the previous homework, Air conditioner and umbrella classes have similar images and also their bow histograms are dominated by the same bins thus these poor training data lead to poor leaning by the SVM.





**Results for Test Data**

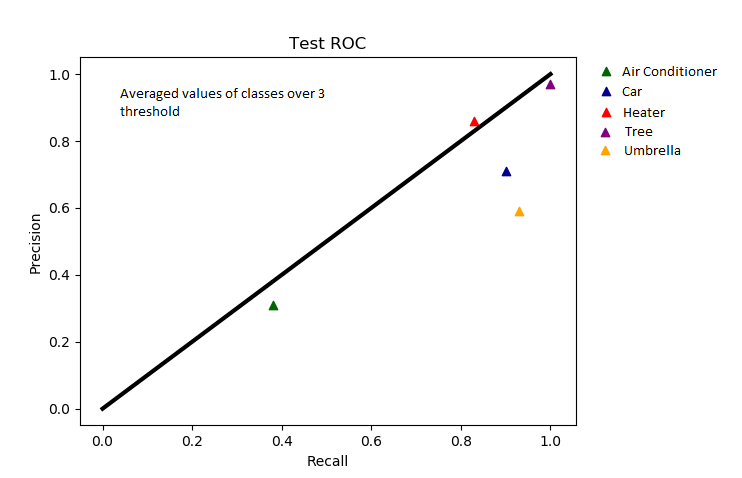
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Threshold = 0** | **Air Conditioner** | **Car** | **Heater** | **Tree** | **Umbrella** |
| **Precision** | 0.3 | 0.73 | 0.88 | 1 | 0.6 |
| **Recall** | 0.25 | 0.8 | 0.7 | 1 | 0.9 |

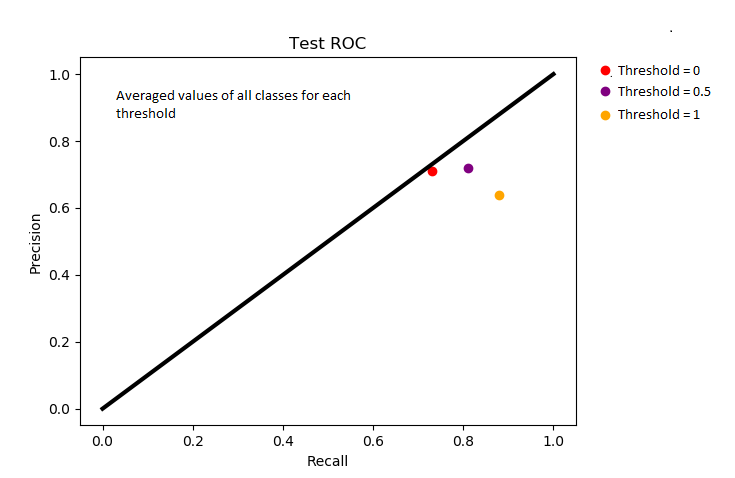
*Every column represents the classifier which trained with positive samples of the respective class name.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Threshold = 0.5** | **Air Conditioner** | **Car** | **Heater** | **Tree** | **Umbrella** |
| **Precision** | 0.33 | 0.75 | 0.90 | 1 | 0.6 |
| **Recall** | 0.38 | 0.9 | 0.9 | 1 | 0.9 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Threshold = 1** | **Air Conditioner** | **Car** | **Heater** | **Tree** | **Umbrella** |
| **Precision** | 0.27 | 0.67 | 0.82 | 0.91 | 0.56 |
| **Recall** | 0.5 | 1 | 0.9 | 1 | 1 |

Considering both above and below data, for the test data, results are similar but expectedly worse. Tree class still has the lead and this time Air Conditioner class by far the worst. These results expected compared to the training data because the model have never seen this test images thus error of the model is increased for test images. Optimum threshold is 0.5. We see that performance of Tree class is the same as training data thus

**

**

**Results of “Bookshelf” Class**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Threshold = 0** | **Air Conditioner** | **Car** | **Heater** | **Tree** | **Umbrella** |
| **TP** | 0 | 0 | 0 | 0 | 0 |
| **TN** | 14 | 10 | 10 | 13 | 15 |
| **FP** | 1 | 5 | 5 | 2 | 0 |
| **FN** | 0 | 0 | 0 | 0 | 0 |

*Every column represents the classifier which trained with positive samples of the respective class name.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Threshold = 0.5** | **Air Conditioner** | **Car** | **Heater** | **Tree** | **Umbrella** |
| **TP** | 0 | 0 | 0 | 0 | 0 |
| **TN** | 14 | 9 | 6 | 12 | 15 |
| **FP** | 1 | 6 | 9 | 3 | 0 |
| **FN** | 0 | 0 | 0 | 0 | 0 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Threshold = 1** | **Air Conditioner** | **Car** | **Heater** | **Tree** | **Umbrella** |
| **TP** | 0 | 0 | 0 | 0 | 0 |
| **TN** | 10 | 7 | 4 | 10 | 13 |
| **FP** | 5 | 8 | 11 | 5 | 2 |
| **FN** | 0 | 0 | 0 | 0 | 0 |

A new class is tested for 3 thresholds. Because of this class wasn’t included in the both training set of vocabulary and the training set of SVM, we expect our classifiers the behave poorly. Results are given above. While our best and worst performed classes gives less false positives for this class, our mediocre classes gave disappointing number of false positives. Because there is no TP and FN, we can’t construct the precision and recall for these data. As we can see, increasing the threshold, meaning, allowing closer distances to hyperplane results in error even in most confident classes.

To determine that these samples are from a new unknown class, we can introduce a new type of function with threshold. We can find each image best match model and then look the distance if it is above a threshold. Although, this would work poorly. We cannot simply look to each distance as the previous threshold or we cannot use the total average distance or a similar approach for it because if we do it this way Heater classifier, most likely will tell that this image is a heater. To achieve this, wee should use a nonlinear relationship function of distances and threshold it or we can improve our training to let simpler approaches to work.

**References**

1. <https://docs.opencv.org/3.4/d1/d73/tutorial_introduction_to_svm.html>
2. <https://en.wikipedia.org/wiki/Receiver_operating_characteristic>
3. EE576 Homework 6

**Fehmi Ayberk Uçkun**

**2015401009**