Project 2: Traffic Sign Recognition

2.1 Traffic Sign Detection

The goal of this section of the project is to extract regions which possibly contains traffic sign. Traffic sign here includes both RED (Danger & Prohibitory) and BLUE (Mandatory) signs which are common to any traffic system. The entire project is done on the RGB color space. The algorithm will be briefed in the pipeline below:

Pipeline:

Processing:

1. The images from the input are sequenced to first separate the red, blue and green channels respectively.
2. The individual channels are then filtered for noise using a median filter(medfilt2).
3. The filtered individual channels are contrast normalized(stretchlim) to adjust the range of pixel intensity values.
4. Then the intensity of the image is normalized using a weighted combination of the three channels that predominantly looks for blue and red signs. The formula for contrast stretching has been modified from the general formulae presented in [1].
5. The formula works well in the images were both blue and red signs are present. For example,

```
blue_sign_1=max(0,max((b_enhanced-g_enhanced-r_enhanced),(r_enhanced-
b_enhanced-g_enhanced)));
```

Fig. 1 Multiple sign detection with the modified formula

MSER:

6. The MSER features are detected in the image with the MATLAB command (detectMSERFeatures) in the Computer Vision Toolbox.
7. Then the region of interest will be measured from the object (mserConnComp) using (regionprops) property.
8. The measured regions are concatenated(vertcat) to put a bounding box over each region.
9. This way, there are some false positives detected along with the original signs. To remove them, we geometrically trim them by varying the following parameters of the bounded regions:

- Aspect ratio > 1.2
- Eccentricity > 0.85
- Solidity < 0.3
- Euler Number < -4

10. The bounding box edge coordinate values are expanded by a small amount to fit the whole sign board instead of just the sign.
11. They are then clipped to fit the image bounds. This eliminates the unnecessary false positives that are not traffic signs. But still, some images do give multiple bounding boxes for the same sign. They overlay on each other.
12. The overlaying problem has been addressed by an overlap ratio setting that simplifies the representation.
13. This approach identifies the connected regions in the representation then merges the bounding boxes based on their end dimensions which is followed by removing the bounding boxes per detected sign.
14. The image is then scaled before highlighting the final bounding box.

Fig. 2 Bounding box for the detected sign

2.2 Traffic Sign Classification

Traffic sign classification follows the traditional approach of detecting the hog features for the image followed by SVM classification.

Pipeline:

1. The input images are evaluated and the class labels are processed before feeding them into hog.
2. During training, the image sets are trained with their appropriate labels and each image of a particular class will be trained.
3. Then the image is processed through the traffic sign detection from the pipeline for 2.1 to extract the bounding boxes to feed the pipeline further.
4. The images are then resized to 64x64 size to fit the hog window size.
5. The hog features for the image are then extracted. The cell size is chosen to be 5, 5 so as to not miss any small-scale detail.
6. The SVM classifier is fed by running test images to fit the multi-class models for the SVM(fitcecoc).
7. This value is then used to predict the multi-class output model.
8. The prediction is then verified by generating a confusion matrix(confusionmat). It computes the confusion matrix that compares the performance for the prediction compared with the targets. In the matrix, the rows represent the true classes and the columns represent the predicted classes.

digit	0	1	2	3	4	5	6	7
0	0.78	0	0.22	0	0	0	0	0
1	0	0.89	0.01	0	0	0	0	0.1
2	0	0	0.99	0	0.01	0	0	0
3	0	0	0.16	0.82	0.01	0	0	0.01
4	0	0	0.01	0	0.95	0.03	0.01	0.01
5	0	0	0	0	0	1	0	0
6	0	0	0	0	0	0	1	0
7	0	0	0	0	0	0	0	1

Fig. 3 Confusion Matrix values after complete classification

9. The bounding box coordinates of the classified image is then used to fit the appropriate image adjacent to the classified label.

Fig. 4 Traffic sign classification with the corresponding image placed near the bounding box

References:

[1] Samuele Salti, Alioscia Petrelli, Federico Tombari, Nicola Fioraio, and Luigi Di Stefano. A traffic sign detection pipeline based on interest region extraction. In Neural Networks (IJCNN), The 2013 International Joint Conference on, pages 17. IEEE, 2013.

[2] Chen, Yixin, Yi Xie, and Yulin Wang. "Detection and Recognition of Traffic Signs Based on HSV Vision Model and Shape features." JCP 8.5 (2013): 1366-1370.

[3] Donoser, Michael, and Horst Bischof. "Efficient maximally stable extremal region (MSER) tracking." Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on. Vol. 1. IEEE, 2006.

[4] https://www.mathworks.com/help/vision/functionlist.html