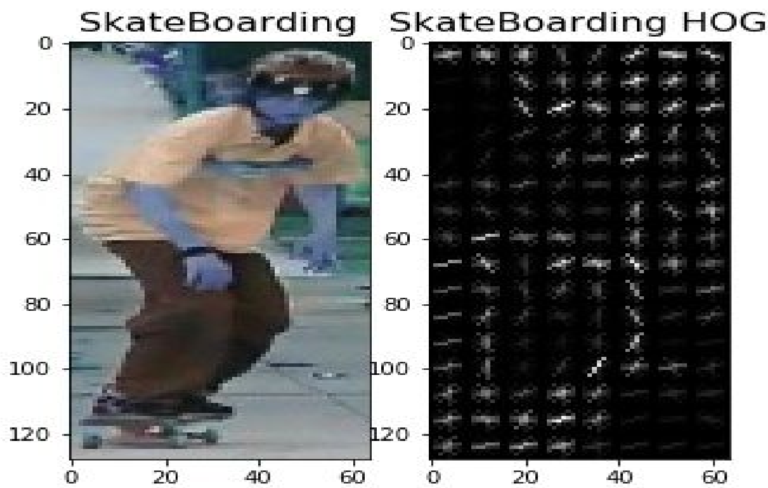
Computer Vision

Programming Assignment -3

Action Recognition - By Rajat Jain

1. **Preprocessing**:
   Before calculating HOG features is necessary to pre- process the image to obtain square-blocks of histograms which are used for a fast HOG calculation. The preprocessing helps feature extraction algorithms to better describe the image features
   So,
   a. Cropping the images according to ground truth given
   b. Resizing the images to 64*128

2. **Feature Extraction**:
   Extracting features before using classifiers-
   a. HOG:Since the HOG descriptor operates on localized cells, the method upholds invariance to geometric and photometric transformations, except for object orientation. Such changes would only appear in larger spatial regions. Moreover, spatial sampling, fine orientation sampling, and strong local photometric normalization permits the individual body movement of human to be ignored so long as they maintain a roughly upright position. The HOG descriptor is thus particularly suited for human detection in images.Feature vector length for my implementation is 3780*3=11340



b. Spatial Features: This gives us segregation between foreground and background of image in color space. Resizing image to 32*32 and adding to feature list.That is 32*32*3=3072 feature vector length

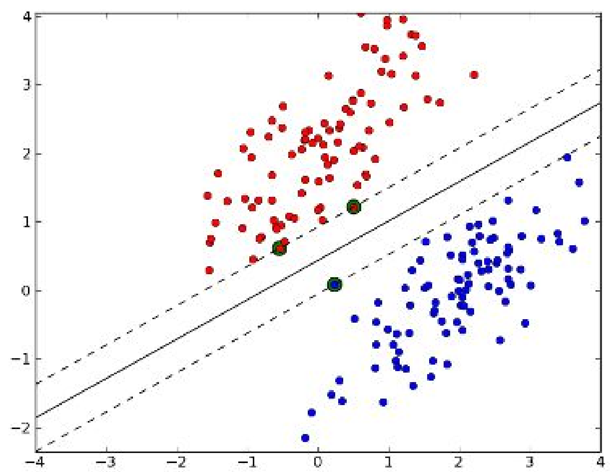
c. Histogram of image:A histogram is computed from all of the pixels in the image, and the peaks and valleys in the histogram are used to locate the clusters in the image. Color or intensity can be used as the measure.Took 32 bins so feature vector length is 32*3=96

Total Feature Vector Length=14508

3. **Classifiers Designs**:

   Tried out with different classifiers

   a. Linear Svm: An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on.



   b. Random Forest: Testing on random forest is fast compare to svm, Used maximum depth as 7

   c. Deep Neural Network: Tried with deep neural network. For this i don’t need feature extraction. I just used preprocessed images and sent it to my keras deep neural network. Training was very fast using this compare to svm.

4. **Leave one out**:
   Testing for this dataset proceeded in a leave-one-out framework. Given the significant intra-class variability present in the sports actions, the recognition task is challenging.
   While running svm training time was very high. Training in random forest is fast so executed it in this. Results are present in evaluation part of report
5. **Train-test split validation**:
   I also executed svm model with train-test-split(Randomly test on 20% data) and received accuracy of 98.7%
6. **Evaluation**:
   Test sensitivity (conditional probability that the test is positive if the condition is positive), calculated by the following formula:
   Sensitivity = (True Positive)/ (True Positive + False Negative) × 100

   Test specificity (conditional probability that the test is normal if the condition is

is

   normal(negative),calculated by following formula
   Specificity = (True Negative)/ (True Negative + False Positive) × 100

	Random Forest	Random Forest Sensitivity	SVM	SVM Sensitivity
	Hog features, Image Hist, Spatial Features		Hog features, Image hist, Spatial Features	
Kicking-Front	0.35	0.35	0.568	0.57
Swing-SideAngle	0.8071	0.8	0.758	0.76
Golf-Swing-Back	0.31	0.31	0.514	0.51
Run-Side	0.7213	0.72	0.7879	0.79
Golf-Swing-Front	0.4578	0.46	0.656	0.66
Swing-Bench	0.567	0.57	0.4522	0.45
Kicking-Side	0.6777	0.68	0.354	0.35
Diving-Side	0.8998	0.9	0.942	0.94
Walk-Front	0.4132	0.41	0.5444	0.54
Riding-Horse	0.578	0.58	0.7401	0.74
Skating Boarding Front	0.8874	0.89	0.904	0.9
Golf-Swing-Side	0.3584	0.36	0.4544	0.45