HW1: Building Video based MED pipeline.

Email : nyarrabe@andrew.cmu.edu

1. **Data**:
   Given 7942 videos of short duration with ten event categories.
   TrainVal set is split into Train set and Validation set using 75-25 split ratio. Stratified split is used to ensure that the class label distribution remains the same for both the train set and validation set.

2. **Feature Sets:**

2.1 SURF :
SURF uses wavelet responses in horizontal and vertical direction for a neighborhood of size 6s. Adequate guassian weights are also applied to it. A neighborhood of size 20sX20s is taken around the keypoint where s is the size. It is divided into 4x4 subregions. Thus we obtain 64 dimensional features for every key point extracted.

Feature dimensions for each video is : # of frames x # of keypoints x # feature dimension (64)
Parameters:
Hessian Threshold : Only features, whose hessian is larger than hessianThreshold are retained by the detector. Therefore, the larger the value, the less keypoints you will get. A good default value could be from 300 to 500, depending from the image contrast.

2.2 CNN features :

AlexNet: The architecture consists of eight layers: five convolutional layers and three fully-connected layers. The fifth convolutional layer is followed by an Overlapping Max Pooling layer, the output of which goes into a series of two fully connected layers. The output of the final fully connected layer is fed into a softmax classifier. Since we are interested in extracting representative features from images, features are extracted from the first fully connected layer, which is a 4096 dimensional feature vector.

E*ach video feature representation is of the order # of frames x 4096.*

ResNet -(18,50,152): Resnet architecture allows for stacking more deeper layers compared to other models. It achieves efficient training performance by adding residual layers to allow for the backpropogations to flow into the initial layers. In the deeper network the additional layers better approximates the mapping than it’s shallower counter part and reduces the error

by a significant margin. Hence it is widely used in all image recognition and identification tasks.
It uses 3*3 filters mostly and then down samples with CNN layers with stride 2. Global average pooling layer and a 100 way FC layer with Softmax at the end. I extracted features from the global pooling layer as it encapsulated all the information present in the image. Resnet-18 has 512 dimesnional features at average pooling layer and 2048 dimensions for Resnet-50 and Resnet-152 architectures.

E*ach video feature representation is of the order # of frames x512.*

3. **Learning Method**:

Fixed dimensional representation for each video:

To get a fixed dimensional representation, I experimented with both mean pooling across all the frames and also using a bag of words based representation with k-means clustering.
k-Means clustering :
Representing a video with bag of words based vector can give a fixed representation. Unlike text, we do not have identities of the features. Hence we use k-means to get a set of representative points.

Mini batched K-means : Using larger representation size will enhance the model generalization. However choosing large number of clusters to get larger feature vectors is not a directly viable option due to memory constraints. To counter these issues, I used mini-batched kmeans with 200 as mini batch size. The overall efficiency of the kmeans may decrease as a result (due to finding a local optima). However it gave better results.
For each video, we pool all the vectors corresponding to the frames and keypoints in it. We get (#of frames*# of Keypoints , 64) dimensional vectors for each video and assign these vectors to learnt clusters and take each cluster as feature dimension and the normalized number of points as feature value.
Number of clusters is determined by shiloute score.

To speed up the cluttering 20% of random sample key-pints are selected for each video are used to train the model. The aggregated train set size for SURF based video features are (689293*64) .

VLAD :
It differs from the BoW image descriptor by recording the difference from the cluster center, rather than the number of SIFTs assigned to the cluster.

Vlad is an extension of BOW based clustering. Instead of using number of descriptors assigned to each cluster it uses the residuals and hence gives #of clusters x #feature dimensions. Hence it gives 256*64 dimensional vector for each descriptor, (where each descriptor is a 256 dimensional vector). Since we have multiple frames for each video. We get (#clusters x #frames x feature dim). I used mean pooling along the frames dimension to get a final #of clusters x #feature dimensions vector. L2 normalization is used along the axis of clusters.

Since the feature is very large I used **PCA at training time** to project the data along 200 principal components for each video.

4. Learning with MLP :

I experimented with MLP learner. It uses a deep neural network architecture for learning the classes.

Layer Depth : Increasing the number of layers increases the model capacity and can learn any complex function. I have used 2 hidden layers for this model. Increasing it too much can overfit on the training data.

Layer Size: Used (200,100). Layer size indicates the number of functions learnt at each layer.

Activation Function : Experimented with tanh and Relu as activation functions.

Dropout ,BatchNorm are also used to make sure the network doesn't regularize.

5. Results

The results are evaluated on the validation set which is 25% stratified split from the train set and hence all the class distributions are maintained same as that of test set.
I experimented mainly with Multi Layer perceptron classifier varying the layer sizes, Regularization thresholds, Non linearity functions.
Hessian Threshold parameter determines number of key-pints captured for each frame. Hence I tried to increase the key-pints captured. However extracting too many key-pints lead to a lo of noise.

Compared to Alexnet's large kernel sizes , Resnet uses 3*3 kernels which are much small and by using multiple kernels the depth of the network increases and hence can learn much complicated features. This gave an increment of 6.3% improvement over Alexnet.

Feature Extraction	Pooling	Accuracy(Validation Set)	Comments
CNN AlexNet (FC2 layer)	Mean pooling : 4096 dimensions	85.51	

Feature Extraction	**Pooling**	**Accuracy(Validation Set)**	**Comments**
RESNET-50	Mean pooling	92.4	
Resnet-152	**MeanPooling**	**94.8**	
CNN ResNet-18(Avg Pooling layer)	Mean pooling : 4096 dimensions	91.6	
SURF	KMeans with 250 clusters	51.49	
SURF	Kmeans with VLAD	57.89	
SURF	Means with 350 clusters	53.45	