Extraction of Interesting Video Fragments from Soccer Matches

Evgeny Toropov, Gaurav Singh, Maheen Rashid

November 7, 2013

1 Introduction

Soccer is probably the most popular televised sport in the world and thousands of soccer matches are contested every weekend. Soccer highlights are posted online on video sharing sites like Youtube, Footytube and these generate tremendous number of hits. However, highlights compilation is done manually and is often tedious. We propose ways to automate soccer highlight video creation by extracting interesting video fragments leveraging the structure of soccer broadcasts such as camera viewpoints, structured filming style and discovered knowledge about the game structure. Key soccer events like a goal are often visually similar to uninteresting non-goal events, and this is the main challenge we need to overcome.

2 Related Work

Our low level processes like shot boundary detection, slow motion detection and shot classification are inspired from the work of Ekin et al [2] but we choose robust algorithms for some of these low level processing tasks. For instance we approach the problem of classifying shots in two ways - First, by calculating spatio-temporal features for video frames and then classifying using Bag of Words and second, by unsupervised clustering of GIST feature responses calculated over representative video frames of shot. In contrast to Ekin et al., we leverage audio volume and pitch to determine video frames of interest.

3 Dataset Used

We use a training dataset comprising of 30 English Premier League 2013-2014 season matches and 300 individual goal clips from Soccer World Cup 2011 as well as earlier League seasons in England and Spain. We automatically obtain goal and other events in full matches by extracting tags from second accurate Match Commentary posted for English Premier League matches on BBC Football website. With the automatic event extractor we can quickly add video highlights to our training set.

4 Input and Expected Outputs

The input of our system would be an unlabeled video of a soccer match with arbitrary length. It's output would be a number of video fragments with fixed length that are detected as interesting. After experimenting we chose the length to be 50 seconds. For each fragment, we will also label it - explaining

why it is interesting. We will label interesting fragments to one of four categories: ”Goal”, ”Full time/Half time”, ”Penalty”, and ”Booking”. These classes may change based on preliminary results.

5 Approach

Match videos have the advantage of being extremely structured in the number and type of video shots that are used. Unconventional viewpoints are rarely used, and the sequence of shots in case of certain match events is extremely predictable.

For example, in the case of a goal, the shots change from a long shot on the goal area, to a closeup of the players (goal keeper, goal scorer celebrating), followed by one or two replays of the goal from different viewpoints, often in slow motion.

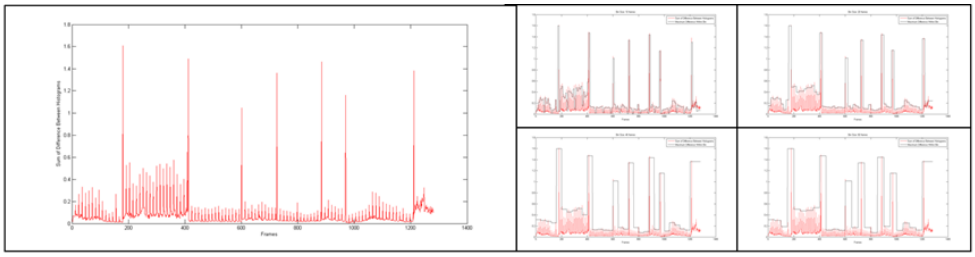
We will use a multi-layer method for classification. Each match video will be split into a number of short video sequences, where each sequence is a separate camera shot. Different features will be separately extracted on each shot. Shots will then be concatenated together to form a fragment that will be classified as either uninteresting, or as one of the categories of interesting.

5.1 Detecting Shot Changes

A camera shot change is characterized by a dramatic change in viewpoint or camera position. We observe that consecutive frames in a video sequence are not likely to be dramatically different from one another unless there is a shot change.

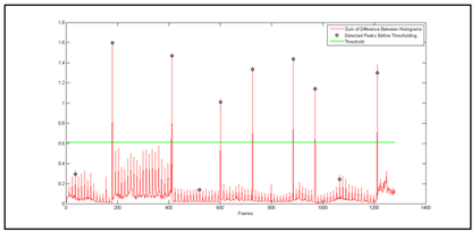
To do this each image is represented with its RGB histograms concatenated together. The sum of differences between consecutive histograms is then taken to form a single number measure of the difference between frames. Fig. 1 (a) is a visualization of the normalized difference between consecutive frames for a short 50 second long clip.

Detecting shot changes requires detecting anomalous peaks in this signal. These signals are not smooth, and shots are of varying lengths. To detect shot changes, we divide the signal into bins. Each bin is assigned the maximum variance of its signal window. This process is repeated for bins with different sizes (we fix bins to be 10,20,30,40, and 50 frames long) . Following, each bin is compared to its neighbouring bins, and a record is kept of bins that are at a local maximum. Finally, the regions of the signal for which bins of all sizes are a local maximum are detected, and the maximum within the smallest sized bin is detected as a peak. Peaks are then thresholded to be greater than half the median value of all detected peaks in a sequence. These peaks are our shot changes. Fig. 1 is a visualization of our shot detection algorithm. Fig. 1 (d) shows some results.



a) Create a Difference of Histograms Signal

b) Bin signals using different sized windows



c) Detect windows that are at a local maxima for all sized bins. Threshold on half the median value. Points above the green line are detected as shot changes

d) Frames from detected shots, ordered left to right, top to bottom.

Figure 1: Detecting Shot Changes (a) shows Difference of Histograms Signal (b) shows Bin signals using different sized windows (c) shows how we detect shot windows (d) shows the final detected shots

5.2 Clustering shots based on Frame Features

As mentioned before, sequences of interesting events often involve the concatenation of shots of certain types stringed together in a certain order. This observation is made in previous work as well [2]. To learn these templates of shots it is necessary to be able to automatically classify shots based on viewpoint, or the subject of the shot.

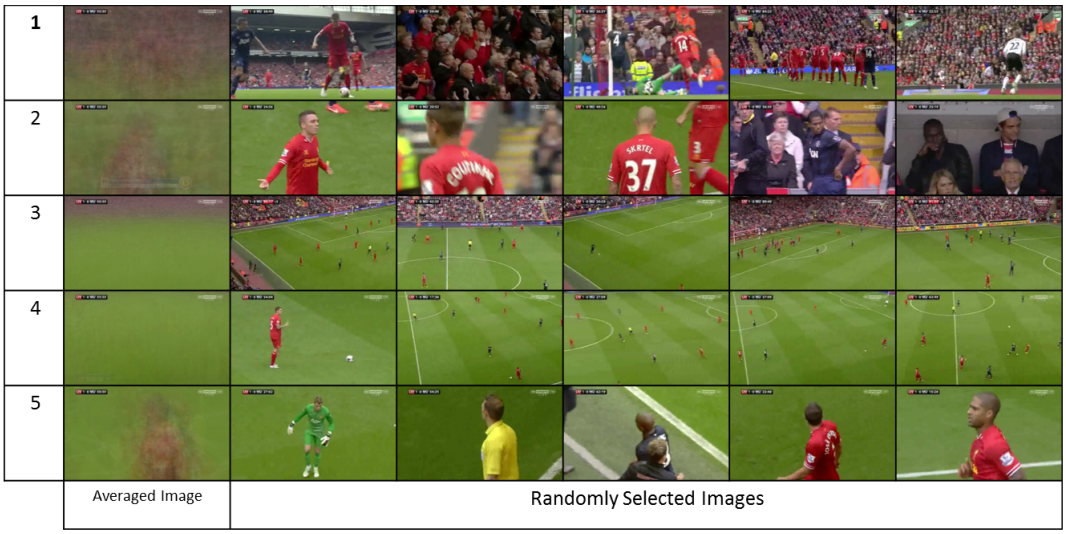


Figure 2: Result of clustering frames from shots for K-means clustering K=5

Since frames within the same shot are visually similar, we represent each shot by a single frame. We then extract GIST features on these images and do K-means clustering on these frames to classify

shots. Fig. 2 shows the average image of 5 clusters and a few randomly chosen images from each cluster. These results indicate that clustering shots can be done quite reliably. However, these results are far from optimal. Ignoring the fact that we might not have enough data (these clusters were formed from 300 images), or enough clusters, or may be using a non-optimal feature representation - there is double counting in the type of clusters that are formed - e.g. cluster 3 and 4 are not qualitatively very different. In addition, cluster membership can be noisy - as can be seen in the results of cluster 2.

For these reasons we will assign shot membership to clusters in a soft manner, and use this as one of the features of each shot.

5.3 Clustering shots based on Video features

As an alternative to the previous method, we will apply the BoW algorithm to cluster shots based on video 2D+time features. We will follow the usual BoW pipeline for video as described, for example, in [4]. We will use HOG/HOF detector and descriptor because it performs the just as well as other detectors, and a little better than other descriptors. We will use the code available at [3].

Following the BoW approach, the descriptors will be clustered, and every cluster will become a visual word. Then the histogram of visual words will be calculated for every shot. After that the pool of shots will be clustered into shot classes.

5.4 Detecting Slow motion in a shot

Slow motion replays are a strong indicator of when an event of interest occurs in a video.

An intuitive approach to detecting slow motion sequences is to compare the rate of change, or motion, in a shot to a non-slow motion sequence. However, detected motion is dependent on viewpoint. A long shot of the field may have as much motion as a slow motion closeup of a goal. Because of this it is necessary to compare slow motion sequences to other sequences taken from the same or similar viewpoint. In context of the previous sections it means - belonging to the same shot clusters.

Confidence that a shot is slow motion is another descriptor we will use for each shot.

5.5 Detecting certain objects in a shot

Some objects may be easy to detect in single frames while they will point to particular events in the fragment. One example is a goal net in large scale (See Fig 3). Because of its periodic structure and consistent color, it may be easy to detected in a frame. At the same time, if the net is shown in large scale then there has probably been a ”hit” or at least, a dangerous situation near the goalpost. These events are definitely of interest in general. The net can detected with horizontal and vertical negative-bimodal filter applied to the whole frame at several quite large scales. For every shot the percentage of time the object is detected is extracted. This value is included into the shot descriptor.

Figure 3: Left image shows Goal net on dark background, Right image shows Goal net on bright background

5.6 Audio Volume and Pitch

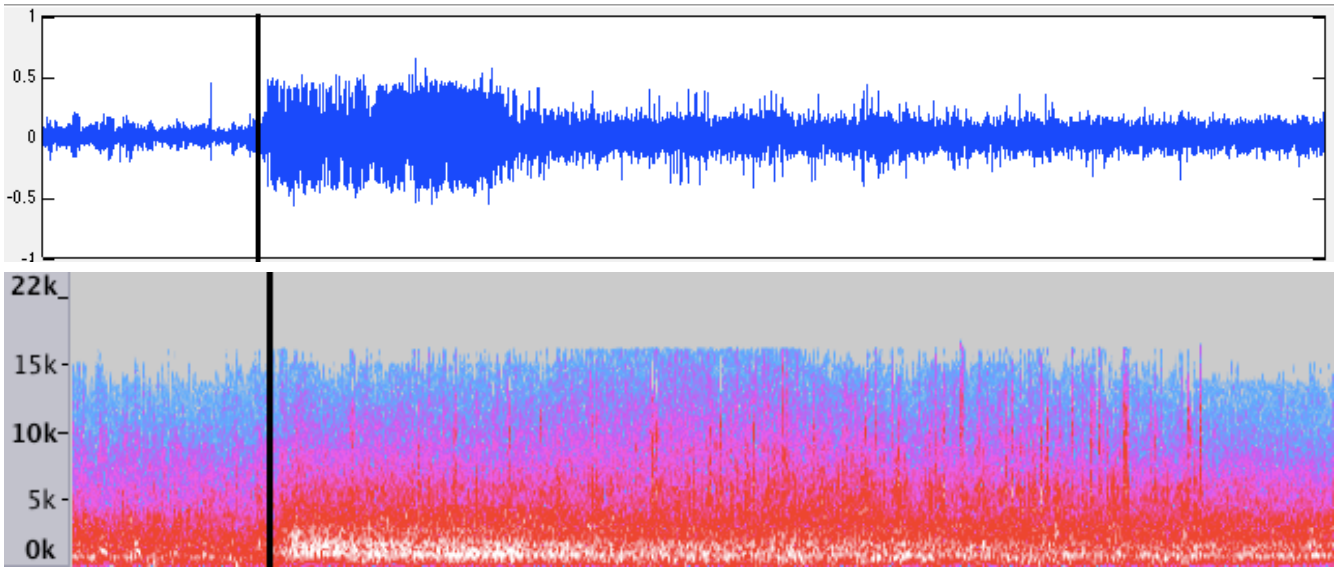


Figure 4: Audio for a 50 second goal fragment TOP volume BOTTOM spectrometer. The black line shows the moment of the goal

It is a known fact the some of commentators are more emotional than others. For a common person that means that the soccer match is more interesting to watch. For this project that also means that the voice of the commentator can be used to get extra information about the fragment. In particular, one would think that the higher volume for the period of 10 seconds would usually mean ”goal” or ”miss”. Higher pitch of the voice can probably be met in any of the events of interest. The faster speech may also signify an interesting event. These assumptions will be checked on real data. Given an unusually high volume and pitch the probability of different events (and absence of event) will be calculated. Accordingly, the

voice volume and pitch will be extracted from a query data segment, and probabilities of different events will be assigned.

6 Getting clues for Classifying Scene from Player Interaction

This section describes the work that can possibly be done given our team has time after completing the main research.

Based on the paper [1] we are planning to explore if it is possible to obtain some extra meaningful information from tracking individual players and recognizing their interactions. For example, a collision of players will probably happen before the ”booking” event.

Every player trajectory will be modelled as a 2D+time subvolume or ”tube” as described in the paper. Relations between tubes will include only ”spatial relation”, ”temporal relations”, and ”merging/splitting”. We will not try to include the ball into the model because the ball is usually very blurred in video frames, and its trajectory on video often intersects trajectories of individual players.

References

[1] William Brendel and Sinisa Todorovic. Learning spatiotemporal graphs of human activities. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 778–785. IEEE, 2011.

[2] A. Ekin, A.M. Tekalp, and R. Mehrotra. Automatic soccer video analysis and summarization. *Image Processing, IEEE Transactions on*, 12(7):796–807, 2003. ISSN 1057-7149. doi: 10.1109/TIP.2003.812758.

[3] Ivan Laptev. Ivan laptev’s page, November 2013. URL `http://www.di.ens.fr/~laptev/download.html`.

[4] Heng Wang, Muhammad Muneeb Ullah, Alexander Klaser, Ivan Laptev, and Cordelia Schmid. Evaluation of local spatio-temporal features for action recognition. In *University of Central Florida, U.S.A*, 2009.