Fast Lead Star Detection in Entertainment Videos

Anonymous CVPR submission

Paper ID 170

Abstract

Video can be summarized in many forms. One natural possibility that has been well explored is extracting key frames in shots or scenes, and then creating thumbnails. Another natural alternative that has been surprisingly ill-explored is to locate "lead stars" around whom the action revolves. Though scarce and far between, available techniques for detecting lead stars is usually video specific.

In this paper, we highlight the importance of lead star detection, and present a generalized method for detecting snippets around lead actors in entertainment videos. Applications that naturally make use of this method include locating action around the 'player of the match' in sports videos, lead actors in movies and TV shows, and guest-host snippets in TV talk shows. Additionally, our method is fifty times faster than the state-of-art spectral clustering technique with comparable accuracy.

1. Introduction

Suppose an avid cricket fan or coach wants to learn exactly how *Flintoff* repeatedly got *Hughes* "out." Or a movie buff wants to watch an emotional scene involving his favourite heroine in a Hollywood movie. Clearly, in such scenarios, you want to skip frames that are "not interesting." One possibility that has been well explored is extracting key frames in shots or scenes and then creating thumbnails. Another natural alternative – the emphasis in this paper – is to determine frames around, what we call, lead stars. A lead star in an entertainment video is the actor who, most likely, appears in many significant frames. We define lead stars in other videos also. For example, the lead star in a soccer match is the hero, or the player of the match, who has scored "important" goals. Intuitively, he is the one the audience has paid to come and see. Similarly the lead star in a talk show is the guest who has been invited, or, for that matter, the hostess. This work presents how to detect lead stars in entertainment videos. Moreover like various video summarization [11, 6, 1], lead stars is a natural way of summarizing video. (Multiple lead stars are of course allowed.)

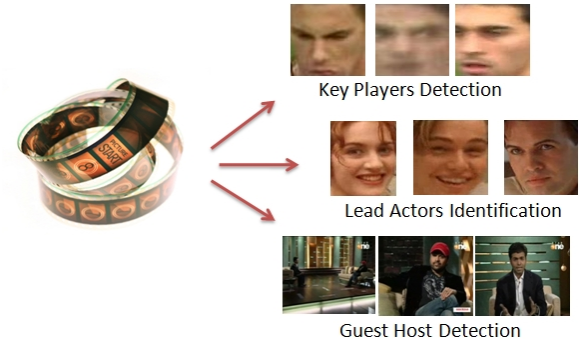

Figure 1. Lead star detection. This is exemplified in sports by the player of the match; in movies, stars are identified; and in TV shows, the guest and host are located.

1.1. Related Work

Researchers have explored various video specific applications for lead stars detection – anchor detection in news video [9], lead casts in comedy sitcoms [2], summarizing meetings [8], guest host detection [6, 5] and locating the lecturer in smart rooms by tracking the face and head [10].

Fitzgibbon [3] uses affine invariant clustering to detect cast listing from movie. As the original algorithm had runtime that is quadratic, the authors used a hierarchical strategy to improve the clustering speed that is central to their method. Foucher, S. and Gagnon, L. [4] used spatial clustering techniques for clustering actor faces. Their methods detect the actor's cluster in unsupervised way with computation time of about 23 hours for a motion picture.

1.2. Our Strategy

Although the lead actor has been defined using a pictorial or semantic concept, an important observation is that the significant frames in an entertainment video is often accompanied by a change in the audio intensity level. It is true no doubt that not all frames containing the lead actors involve significant audio differences. Our interest is not at the frame level, however. Note that certainly the advent of

important scenes and important people bear a strong correlation to the audio level. We surveyed around one hundred movies, and found that it is rarely, if at all, the case that the lead star does not appear in audio highlighted sections, although the nature of the audio may change from scene to scene. And as alluded above, once the lead star has entered the shot, the frames may well contain normal audio levels.

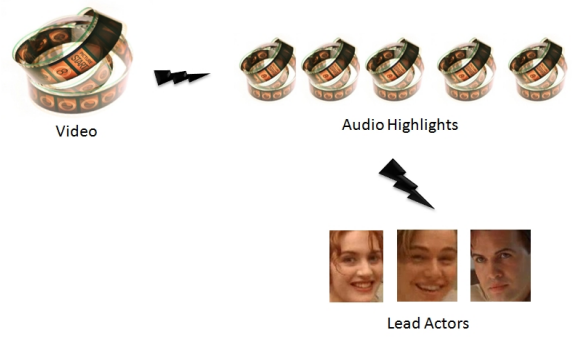

Figure 2. Our strategy for lead star detection. We detect lead stars by considering segments that involve significant change in audio level. However, this by itself is not enough!

Our method is built upon this concept. We detect lead stars considering such important scenes of the video. To reduce false positives and negatives, our method clusters the faces for each important scenes separately and then combines the results. Unlike the method in [3], our method provides a natural segmentation for clustering. Our method is shown to considerably reduce the computation time of the previously mentioned state-of-the-art for computing lead star in motion picture (a factor of 50). We apply this method to sports video to identify *the player of the match*, motion pictures to find *heroes and heroines* and TV show to detect *guest and host*.

2. Methodology

As mentioned, the first step in the problem is to find important scenes which have audio highlights. Once such important scenes are identified, they are further examined for potential faces. Once a potential face is found in a frame, subsequent frames are further analyzed for false alarms using concepts from tracking. At this point, several areas are identified as faces. Such confirmed faces are grouped into clusters to identify the lead stars.

2.1. Audio Highlight Detection

The intensity of a segment of an audio signal is summarized by the root-mean-square value. The audio track of a video is divided into windows of equal size and the *rms* value is computed for each audio window. From the resulting *rms* sequence, the *rms ratio* is computed for successive items in the sequence.

Figure 4. Illustration of highlight detection from audio signal. We detect highlights by considering segments that involve significant low RMS ratio.

The *rms ratio* is marked as low when the value is below a user defined threshold. In our implementation, we use 5 as the threshold, and the video frames corresponding to such windows are considered as 'important.'

2.2. Finding & Tracking Potential People

Figure 5. Illustration of data matrix formation. The tracked faces are stacked together to form the data matrix.

Once important scenes are marked, we seek to identify people in the corresponding video segment. Fortunately there are well understood algorithms that detect faces in an image frame. We select a random frame within the window and detect faces using the Viola & Jones face detector [7].

Every face detected in the current frame is then voted for a confirmation by attempting to track them in subsequent frames in the window. Confirmed faces are stored for the next step in the processing in a data matrix. Confirmed faces from each highlight i, is stored in the corresponding data matrix D_i as illustrated in Figure 5.

2.3. Face Dictionary Formation

In this step, the confirmed faces are grouped based on their features. There are a variety of algorithms for dimen-

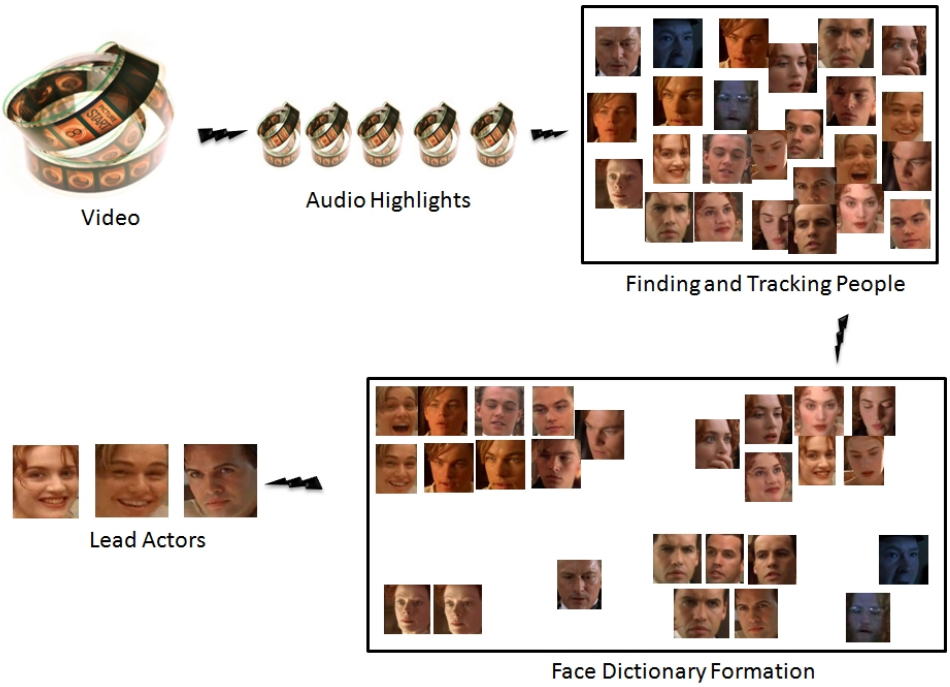

Figure 3. Strategy further unfolded. We first detect scenes accompanied by change in audio level. Next we look for faces in these important scenes, and to further confirm the suitability track faces in subsequent frames. Finally, a face dictionary representing the lead stars is formed by clustering the confirmed faces.

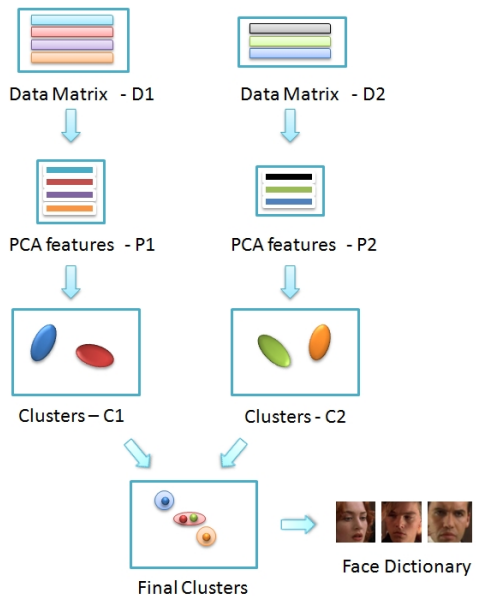

Figure 6. Illustration of face dictionary formation.

sionality reduction, and subsequent grouping. We observe that the Principal Component Analysis (PCA) method has been successfully used for face recognition. We use PCA to extract feature vectors from D_i and we use the k-means algorithm for clustering. The number of clusters is decided based on minimal mean square error. Representative faces from clusters of all highlights are clustered again to get the final set of clusters. The representative faces of these clusters forms the face dictionary.

At this point, we have a dictionary of faces, but not all faces belong to lead actors. We use the following parameters to shortlist the faces to form lead stars.

1. The number of faces in the cluster. If a cluster (presumably of the same face) has a large cardinality, we give this cluster a high weightage.

2. Position of the face with respect to center of the image. Lead stars are expected to be in the center of the image.

3. Size of the detected face. Again, lead stars typically occupy a significant portion of the image.

4. Duration for which the faces in the cluster occur in the current window as a fraction of the window size.

The face dictionary formed for the movie *Titanic* is shown in Figure 7. Our method has successfully detected the lead actors in movie. As can be noticed, along with lead stars, there are patches that have been misclassified as faces.

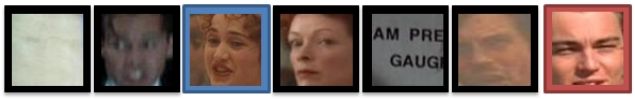

Figure 7. Face dictionary formed for the movie *Titanic*. Lead stars are highlighted in blue (third image), and red (last image). Note that this figure does not indicate the frequency of detection.

3. Applications

In this section, we demonstrate our technique using three applications — *Lead Actor Detection in Motion Picture*, *Player of the Match Identification* and *Host Guest Detection*. As applications go, *Player of the Match Identification* has not been well explored considering the enormous interest. In the other two applications, our technique detects lead stars faster than the state-of-art techniques, which makes our method practical and easy to use.

3.1. Lead Actor Detection in Motion Picture

In motion pictures, detecting the hero, heroine and villain has many interesting benefits. A person while reviewing a movie can skip the scenes where lead actors are not present. A profile of the lead actors can also be generated. Significant scenes containing many lead actors can be used for summarizing video.

In our method, the face dictionary formed contains the lead actors. These face dictionaries are good enough in most of the cases. However, for more accurate results, the algorithm scans through a few frames of every shot to determine the faces occurring in the shot. The actors who occur in a large number of shots is identified as the lead actor. The result of lead actors for the movie *Titanic* after scanning through entire movie is shown in the Figure 8.

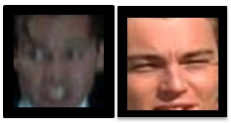

Figure 8. Lead actors detected for the movie *Titanic*

.

3.2. Player of the Match Identification

In the sports, sports highlight and key frames [6] are the two main methods used for summarizing. We summarize sports using the *player of the match* capturing the star players.

Detecting and tracking players in the complete sports video does not yield player of the match. The star players can play for shorter time and score more as opposed to players who attempt many times and don't. So analyzing the players when there is score leads to the identification of star players. This is easily achieved by our technique, as detecting highlights results in exciting scenes like scores.

Figure 9. Lead star detected for the highlights of a soccer match *Liverpool vs Havant & Waterlooville*. The first image is erroneously detected as face. The other results represents players and coach.

The result of lead sports stars detected from a soccer match *Liverpool vs Havant & Waterlooville* is presented in the Figure 9. The key players of the match are detected.

3.3. Host Guest Detection

In TV interviews and other TV programs, detecting host and guest of the program is the key information used in video retrieval. Javed et. al. [5] have proposed a method for the same which removes the commercials and then exploits the structure of the program to detect guest and host. The algorithm uses the inherent structure of the interview that the host occurs for shorter duration than guest. However, it is not always the case, especially when the hosts are equally popular like in the case of TV shows like *Koffee With Karan*. In the case of competition shows, the host is shown for longer duration than guests or judges.

Our algorithm detects hosts and guests as lead stars. To distinguish hosts and guests, we detect lead stars on multiple episodes and combine the result. As it is intuitive, the lead stars over multiple episodes are hosts and the other lead stars detected for specific episodes are guests.

4. Experiments

We have implemented our system in Matlab. We tested our method on an Intel Core Duo processor, 1.6 Ghz, 2GB RAM. We have conducted experiments on 7 popular motion pictures, 9 soccer match highlights and two episodes of TV shows summing up to total of 19 hours 23 minutes of video. Our method detected lead stars in all the videos in an average of 14 minutes for an one hour video. The method [4] in the literature computes lead star of a motion picture in 23 hours, whereas we compute lead star for motion picture in an average of 30 minutes. We now provide more details.

4.1. Lead Actor Detection in Motion Picture

We ran our experiments on 7 box-office hit movies listed in the table 1. This totally sums up to 16 hours of video.

Table 1. Time taken for computing lead actors for popular movies.

No.	Movie Name	Duration (hh:mm)	Computation Time for Detecting Lead Stars (hh:mm)	Computation Time for Refinement (hh:mm)
1	The Matrix	02:16	00:22	00:49
2	The Matrix Reloaded	02:13	00:38	01:25
3	Matrix Revolutions	02:04	00:45	01:07
4	Eyes Wide Shut	02:39	00:12	**00:29**
5	Austin Powers in Goldmember	01:34	**00:04**	01:01
6	The Sisterhood of the Traveling Pants	01:59	00:16	00:47
7	Titanic	03:18	**01:01**	**01:42**
	Total	16:03	03:18	07:20

Figure 10. Lead actors identified for popular movies appear on the right.

The lead stars in all these movies are computed in 3 hour 18 minutes. So the average computation time for a movie is around 30 minutes. From Table 1, we see that the best computation time is 4 minutes for the movie *Austin Powers in Goldmember* which is 1 hour 42 minutes in duration. The worst computation time is 45 minutes for the movie Matrix Revolutions of duration 2 hour 4 minutes. For movies like *Eyes Wide Shut* and *Austin Powers in Goldmember*, the computation is faster as there are fewer audio highlights. Whereas action movies like *Titanic* sequels take more time as there are many audio highlights. This causes the variation in computation time among movies.

The lead actors detected are shown in the Figure 10. The topmost star is highlighted in red color and the next top star is highlighted in blue color. As you can notice in the figure, in most of the movies topmost stars are detected. Since the definition of "top" is subjective, it could be said that in some cases, top stars are not detected in some cases. Further, in some cases the program identifies the same actor multiple times. This could be due to disguise, or due to pose variation. The result is further refined for better accuracy as mentioned in Section 3.1.

4.2. Player of the Match Identification

We have conducted experiments on 11 soccer match highlights taken from BBC and listed in Table 2. Our method on an average takes half the time of the duration of the video. Note however, that these timings are for only sections that have already been manually edited by the BBC staff. If the video were run on a routine full soccer match,

we expect our running time to be a lower percentage of the entire video.

Table 2. Time taken for computing key players from BBC MOTD highlights for premier league 2007-08.

No.	Soccer Match	Duration (hh:mm)	Computation (hh:mm)
1	Barnsley vs Chelsea	00:02	00:01
2	BirminghamCity vs Arsena	00:12	00:04
3	Tottenham vs Arsenal	00:21	00:07
4	Chelsea vs Arsenal	00:14	00:05
5	Chelsea vs Middlesborough	00:09	00:05
6	Liverpool vs Arsena	00:12	00:05
7	Liverpool vs Havant & Waterlooville	00:15	00:06
8	Liverpool vs Middlesbrough	00:09	00:05
9	Liverpool vs NewcasIteUnited	00:18	00:04
	Total	01:52	00:43

The results of key player detection is presented in the Figure 11. The key players of the match are identified for all the matches.

4.3. Host Guest Detection

We conducted our experiment on the TV show *Koffee with Karan*. Two different episodes of the show were combined and fed as input. Our method identified the host in 4 minutes for a video of duration 1 hour 29 minutes. Our method is faster than the method proposed by Javed et. al. [5].

Table 3. Time taken for identifying host in a TV show *Koffee With Karan*. Two episodes are combined into a single video and given as input.

TV show	Duration (hh:mm)	Computation (hh:mm)
Koffee With Karan	01:29	00:04

The result of our method for the TV show *Koffee with Karan* is presented in Figure 12. Our method has successfully identified the host.

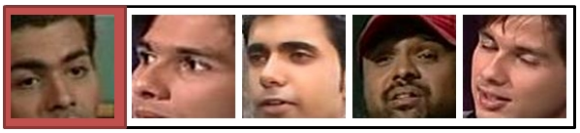

Figure 12. Host detection of TV show "Koffee with Karan". Two episodes of the show are combined and given as input. The first person in the detected list (sorted by the weight) gives the host.

5. Conclusion

Detecting lead stars has numerous applications like identifying player of the match, detecting lead actor and actress in motion pictures, guest host identification. Computational time has always been a bottleneck for using this technique. In our work, we have presented a faster method to solve this problem with comparable accuracy. This makes our algorithm usable in practice.

References

[1] A. Doulamis and N. Doulamis. Optimal content-based video decomposition for interactive video navigation. *Circuits and Systems for Video Technology, IEEE Transactions on*, 14(6):757–775, June 2004.

[2] M. Everingham and A. Zisserman. Automated visual identification of characters in situation comedies. In *ICPR '04: Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04) Volume 4*, pages 983–986, Washington, DC, USA, 2004. IEEE Computer Society.

[3] A. W. Fitzgibbon and A. Zisserman. On affine invariant clustering and automatic cast listing in movies. In *ECCV '02: Proceedings of the 7th European Conference on Computer Vision-Part III*, pages 304–320, London, UK, 2002. Springer-Verlag.

[4] S. Foucher and L. Gagnon. Automatic detection and clustering of actor faces based on spectral clustering techniques. In *Proceedings of the Fourth Canadian Conference on Computer and Robot Vision*, pages 113–122, 2007.

[5] O. Javed, Z. Rasheed, and M. Shah. A framework for segmentation of talk and game shows. *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, 2:532–537 vol.2, 2001.

[6] Y. Takahashi, N. Nitta, and N. Babaguchi. Video summarization for large sports video archives. *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*, pages 1170–1173, July 2005.

[7] P. Viola and M. Jones. Robust real-time face detection. *International Journal of Computer Vision*, 57(2):137–154, May 2004.

Figure 11. Key players detected from BBC *Match of the Day* match highlights for premier league 2007-08.

[8] A. Waibel, M. Bett, and M. Finke. Meeting browser: Tracking and summarizing meetings. In *Proceedings DARPA Broadcast News Transcription and Understanding Workshop*, pages 281–286, February 1998.

[9] D. Xu, X. Li, Z. Liu, and Y. Yuan. Anchorperson extraction for picture in picture news video. *Pattern Recogn. Lett.*, 25(14):1587–1594, 2004.

[10] Z. Zhang, G. Potamianos, A. W. Senior, and T. S. Huang. Joint face and head tracking inside multi-camera smart rooms. *Signal, Image and Video Processing*, 1:163–178, 2007.

[11] Y. Zhuang, Y. Rui, T. Huang, and S. Mehrotra. Adaptive key frame extraction using unsupervised clustering. In *Proceedings of the International Conference on Image Processing*, volume 1, pages 866–870, 1998.

Appendix: Algorithm

Algorithm for detecting lead stars is presented below. In motion pictures, this indicates the lead actors. In sports, this fetches the key players. In TV show, among episodes, fetches host as the top most star.

Detection of lead actors

```
program LeadActorDetection (Video)
 var
  Highlights: Int[][];
  Faces, FacesInInterval: List<Image>;
  Interval, Shots, RandomFrames: int[];
  LeadActorsClusterCenters: List<Image>;
  LeadActors: List<Image>;
begin
  Highlights := GetAudioHighLights(Video);
  Faces := [];
  i := 0
  repeat
   Interval := Highlights(i);
   Shots := ComputeShot(Video, Interval);
   RandomFrame := GetRandomNumber(Interval);
   FacesInInterval := ExtractFace(Video,
                      RandomFrame, Shots);
   Faces.add(FacesInInterval);
   i := i + 1;
  until i <= length(Highlights)
  LeadActorsClsCenters := Cluster(Faces);
  LeadActors := SelectTopClusters(
                LeadActorsClsCenters, Faces);
end.
```