A Fuzzy Theoretic Approach for Camera Motion Detection

R.S. Jadon
CSE Dept.
IIT Delhi
New Delhi-16
rsjadon@cse.iitd.ernet.in

Santanu Chaudhury
EE Dept.
IIT Delhi
New Delhi-16
santanuc@ee.iitd.ernet.in

K.K. Biswas
CSE Dept.
IIT Delhi
New Delhi-16
kkb@cse.iitd.ernet.in

Abstract

Camera motion detection is an important activity for video data processing. The complexity lies in distinguishing real camera movements and object motions. We have proposed a fuzzy theoretic framework for detecting and categorizing camera motion in video clips. The output of the optical flow computation is fuzzified for the purpose of camera motion detection. We have detected and identified the zoom and pan motions in video sequences.

1 Introduction

Information about camera operation is very important for the analysis and classification of video shots, since camera operation often reflects the intentions of the director [3]. There are two important camera operations: panning and zooming. Each of these operations induces a specific pattern in the field of motion vectors from one frame to the next. In this paper,we have proposed a fuzzy theoretic approach for qualitative characterization of camera motion in a video sequence.

Camera work analysis was attempted using motion vector field analysis in [8], [1], [10]. The results obtained by these methods for direct global estimation from MPEG-type motion vectors are not very good. Edoardo et al. [2] have suggested a video indexing scheme using optical flow fields. The optical flow is computed only from few r-frames and its adjacent frames. Sudhir and Lee [7] divided optical flow into singular and non-singular according to whether or not the optical flow vanishes at the camera center. The singular flow is sub-classified into Z-rotation and Z-translation zoom by using an affine transform. The non-singular flow is sub-classified into camera translations and rotations by computing the magnitude of the observed optical flow vectors. Srinivasan et al. [6] observed that the residual optic flow vectors were parallel(Z-translation is omitted) when the components of the optic flow, due to camera rotation and zoom, were subtracted. They used an iterative algorithm to minimize deviation from parallelism of the residual flow vectors. They found r_x, r_y, r_z and r_{zoom} to be the best estimate of tilt, pan, roll and zoom.

These schemes, proposed for characterization of camera motion, are based upon crisp estimation of motion parameters. However, due to variations in the imaging conditions, object motion in the scene and innovative usages of camera motion in video, crisp estimates do not always provide a robust mechanism for qualitative assessment of the type of camera motion. This has motivated us to propose a fuzzy theoretic approach to analyze attributes of motion vectors which can appropriately detect the pan/tilt and zoom camera motion. Our scheme uses a fuzzy rule based system for robust categorization of camera motion in video sequences.

The rest of the paper is organized as follows: In section 2 we will discuss the pan detection scheme. In section 3 the zoom detection technique is presented. The results of case studies are presented in section 4. Finally we conclude in section 5.

2 Pan Detection

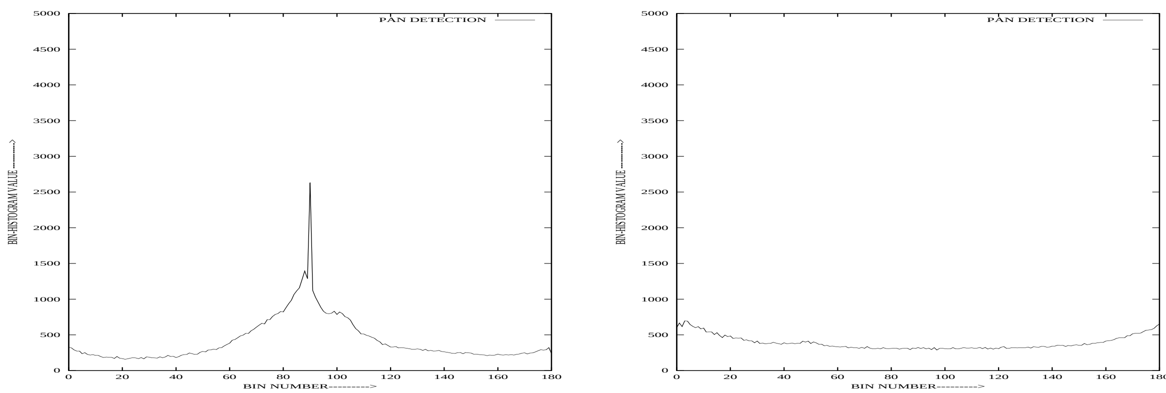
Camera movements can be of different types. For our work we have grouped horizontal and vertical camera translation and vertical and horizontal swiveling of the camera about the same base position (horizontal track, vertical track, following pan, tilting) into the broad category

Figure 1: Motion Vectors: Panning-Shot

of panning. All these types of camera motion provides an incremental coverage of space. A simple translational camera movement is shown in Fig. 1 using the needle diagram for representing the optic flow vectors. This class of camera motion, as shown in the figure, can be detected by estimating the direction of optical flow vector. Horn and Shunk's algorithm [4] has been used for this purpose. It gives motion vectors u and v in x-direction and y-direction respectively. The direction of motion at each pixel in the image is obtained by:

$$\theta = atan(u/v)$$

We then compute the normalized histogram of the angular values, over the range $-\pi/2$ to $+\pi/2$(180 values) for each pair of consecutive frames. We compute the median histogram from these histograms. This is used as the key-feature for characterizing a panning sequence. A typical distribution of angular values for a panning and a non-panning shots is shown in Fig. 2. As can be seen in the figure, there is a prominent peak for the panning shot, while the distribution is relatively uniform for the non-panning shots. The fast-panning shots resulting in unimodal direction histogram can be detected by simple motion estimation algorithms using exact reasoning [9]. However the situation becomes complex with slow-panning shots where the dominant direction is not very apparent. Further, there are problems when there is large object motion in the shot. To deal with such variations, we propose a fuzzy logic based scheme. The scheme involves fuzzification at two levels. At the first level the histogram frequencies are fuzzified while at the second level the histogram bin count is fuzzified.



(a) (b)

Figure 2: Phase Histogram: (a)Panning Shot (b) Non-panning Shot

2.1 Fuzzy Scheme for Pan Detection

At the first level the normalized direction histogram values for each bin are fuzzified as *small, large and very-large.* The fuzzy boundaries for these classes are determined statistically as percentage of total number of pixels. These labels are used to characterize the bins of the histogram as *Most-Significant, Significant and Least-Significant.* We then apply alpha-cuts to the fuzzy sets **most-significant-bin, significant-bin and least-significant-Bin**. The resulting fuzzy-sets are represented as $\alpha_1 MSB$, $\alpha_2 SB$ and $\alpha_3 LSB$. The alpha values used in our experimental system were 0.1 for α_1MSB, 0.2 for α_2SB and 0.8 for α_3LSB. The membership functions for these sets are shown in Fig. 3. The cardinality of these sets is used as input at the second level of fuzzification. We denote these cardinalities as **cdn(α_1MSB), cdn(α_2SB), cdn(α_3LSB)**. These are further fuzzified as **nearly-one, few and large**. Another feature which is important for detecting panning-effects is **span** of a set. We define it to be the diameter of these alpha-cut sets. This feature is fuzzified as *small, large and very-large.* The membership functions for these sets are shown in Fig. 4. Using these fuzzy predicates we have formulated rules exploiting the basic heuristic that *in a panning-shot large number of pixels tend to move in the same direction.* Some of the fuzzy rules, for final characterization of panning and non-panning shots are listed below:

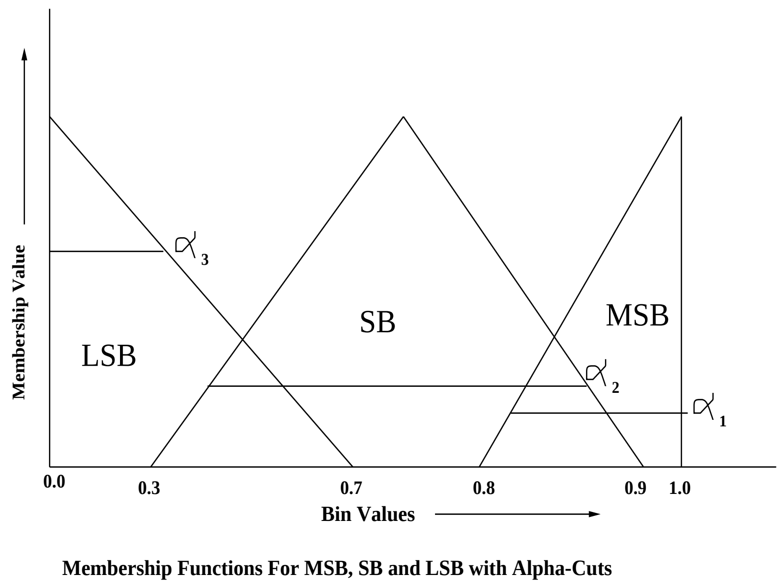


Figure 3: Membership Functions for bin-value characterization in Pan Detection

- **If** $cdn(\alpha_1 MSB)$ **is nearly-one and span of** $\alpha_1 MSB$ **is small and** $cdn(\alpha_3 LSB)$ **is large then it is** $PANNING-SHOT$
- **If** $cdn(\alpha_1 MSB)$ **is few and span of** $\alpha_1 MSB$ **is small and** $cdn(\alpha_3 LSB)$ **is large then it is a** $PANNING-SHOT$
- **If** $cdn(\alpha_1 MSB)$ **is few and span of** $\alpha_1 MSB$ **is large and** $cdn(\alpha_3 LSB)$ **is large then it is a** $NON-PANNING-SHOT$
- **If** $cdn(\alpha_1 MSB)$ **is large and** $cdn(\alpha_3 LSB)$ **is few then it is a** $NON-PANNING-SHOT$
- **If** $cdn(\alpha_2 SB)$ **is large and** $cdn(\alpha_3 LSB)$ **is large then it is a** $NON-PANNING-SHOT$
- **If** $cdn(\alpha_2 SB)$ **is** $large$ **and** $cdn(\alpha_1 MSB)$ **is large then it is** $NON-PANNING-SHOT$

Intuitively the rules are taking care of the fact that in a panning-shot there will be few dominant bins. If the number of such bins is one than this will be an ideal panning-shot. The panning shots are detected with certain membership value. The higher membership value is positive indication for panning.

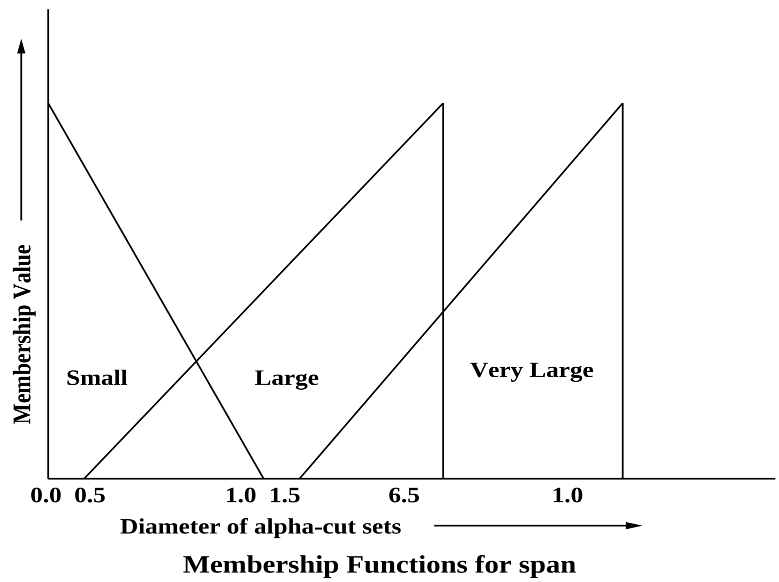


Figure 4: Membership Functions for span

3 Zoom Detection

In Zoom-in the subject is gradually magnified as the lens is focussed down from a long-shot to a close-up. Zoom-out reveals more of the scene as the shot widens. Camera actually remains static in both the cases. However, in many situations camera is dollied in or out coupled with zooming. It is well known that the optic flow velocity gradient tensor can be decomposed into the first order differential invariants of the image velocity field: The curl, divergence and the pure deformation. The use of divergence for collision detection is reported in [5]. In this paper we have proposed the use of divergence for detecting and classifying the zooming effects. To a first order expansion, the image velocity field in a small field of view around the direction of view can be described by:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} u_o \\ v_o \end{bmatrix} + \begin{bmatrix} u_x & u_y \\ v_x & v_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Where (u_0, v_0) is the image velocity and u_x, u_y, v_x, v_y are the partial derivatives of u,v with respect to the indicated subscripts x,y. The above expression represents an affine transformation. In this expression, the $2X2$ tensor on the right hand side is the velocity gradient tensor. The divergence is given by the trace of this velocity gradient tensor:

$$divergence = u_x + v_y$$

The sign of divergence indicates whether the flow of optical energy is towards or away from the focus of expansion. If the sign is positive it is zoom-in. On the other hand, if it is negative we observe zoom-out. We illustrate this effect with the help of needle diagram in Fig. 5.

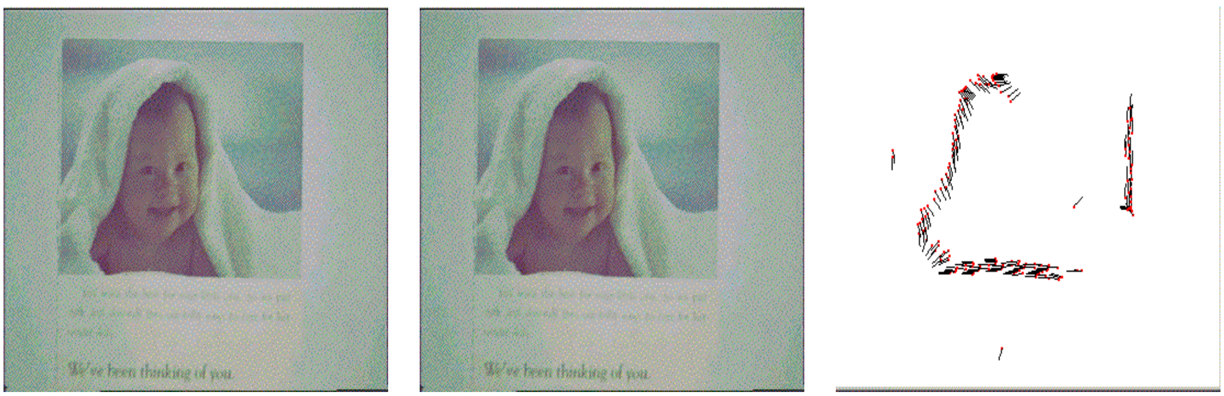


Figure 5: Motion Vectors: Zooming-Shot

We have developed a two level fuzzification scheme for detecting zoom-in, zoom-out and non-zoom sequences using the positive and negative divergence values. At first the individual divergence values are fuzzified as: *high-positive, medium-positive, low-positive, high-negative, medium-negative and low-negative.* These are computed for each pair of frames, resulting in six fuzzy sets for the entire shot. We then apply alpha-cuts to these sets. The resulting fuzzy-sets are represented as $\alpha - hp$, $\alpha - mp$, $\alpha - lp$, $\alpha - hn$, $\alpha - mn$ *and* $\alpha - ln$. The actual alpha value used in our experimental system is 0.6. The membership function for these sets are shown in Fig. 6. The cardinality of these sets is used as input to the second level of fuzzification. These are represented as: $cdn(\alpha - hp)$, $cdn(\alpha - mp)$, $cdn(\alpha - lp)$, $cdn(\alpha - hn)$, $cdn(\alpha - mn)$ **and** $cdn(\alpha - ln)$. These are fuzzified into two categories *small and large.* Fuzzy rules for zoom-detection are motivated by the following heuristics: If highly positive divergence values are large and highly negative divergence values are small then it is zoom-in sequence, and conversely if highly negative divergence values are large and highly positive divergence values are small then it is zoom-out sequence.

Some of the typical rules for final zoom detection are listed below:

- **If** $cdn(\alpha - hp)$ **is large and** $cdn(\alpha - ln)$ **is small then it is zoom-in sequence**
- **If** $cdn(\alpha - hp)$ **is large and** $cdn(\alpha - lp)$ **is small then it is zoom-in sequence**
- **If** $cdn(\alpha - lp)$ **is large and** $cdn(\alpha - mn)$ **is small then it is zoom-in sequence**
- **If** $cdn(\alpha - hn)$ **is large and** $cdn(\alpha - lp)$ **is small then it is zoom-out sequence**
- **If** $cdn(\alpha - mn)$ **is large and** $cdn(\alpha - lp)$ **is small then it is zoom-out sequence**
- **If** $cdn(\alpha - ln)$ **is large and** $cdn(\alpha - hn)$ **is small then it is zoom-out sequence**
- **If** $cdn(\alpha - hn)$ **is large and** $cdn(\alpha - mp)$ **is small then it is zoom-out sequence**
- **If** $cdn(\alpha - mp)$ **is large and** $cdn(\alpha - mn)$ **is small then it is non-zoom**
- **If** $cdn(\alpha - lp)$ **is large and** $cdn(\alpha - ln)$ **is large then it is non-zoom**
- **If** $cdn(\alpha - lp)$ **is small and** $cdn(\alpha - ln)$ **is small and** $cdn(\alpha - hp)$ **is small then non-zoom**

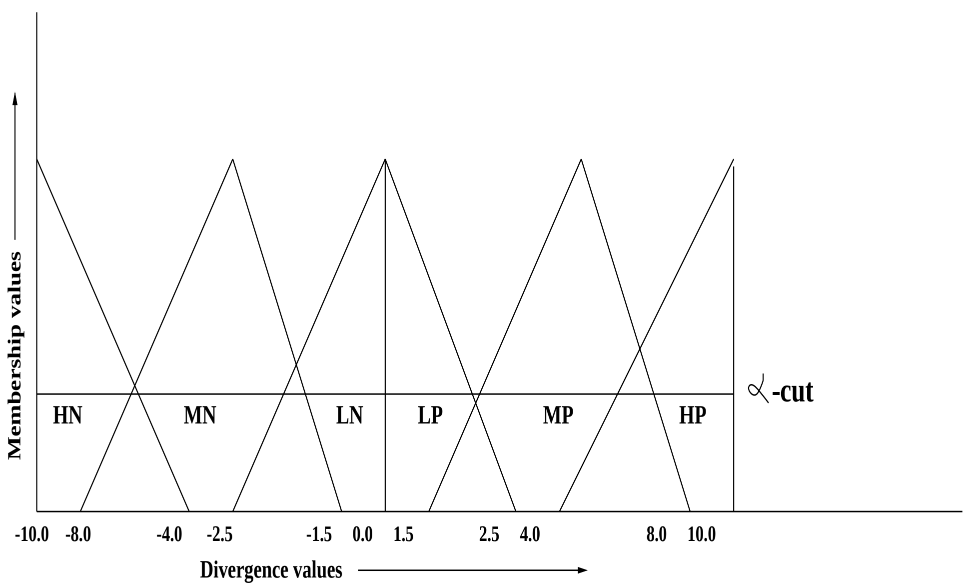


Figure 6: Membership Functions for Zoom Detection

4 Implementation and Experimental Results

We have implemented the system on SGI(IRIX 6.3) workstations. We have done experimentation with about 100 video sequences comprising of news and sports clips, documentaries and feature films. A typical sub-sampled panning shot is shown in Fig. 7. In this shot the camera is moved around a round table to capture the happenings of conference. The distribution of angular values for this shot is shown in Fig. 8(a). Despite being a multi-modal histogram it has been correctly identified as a panning shot with membership value of 0.78 because of our fuzzy rules. The distribution of angular values for a slow-panning shot of Fig. 9 is shown in Fig. 8(b). Our scheme could detect such panning shots also, but with a lower membership value(0.51 for this sequence). In this shot significant-bins are in category *few*, due to which, its membership-value in panning-shot category is less.

A sub-sampled zoom-in sequence is shown in Fig. 10. It is a cricket shot with camera zoomed-in to show the batting action of the player while bowler is bowling from the camera's end. In this shot there are 146 frames. The divergence value is obtained highly-positive for 130 frames and low-negative for 16 frames. Thus the positive divergence component is very high as compared to negative divergence component, resulting in a zoom-in sequence with membership value 0.81.

We have presented the detailed results for camera motion detection in Table- 1. The panning-module gives fuzzy membership value in two categories: pan and non-pan. The maximum of these is inferred as the overall panning motion of the shot. Similarly the maximum of zoom-in, zoom-out and non-zoom is inferred as the output of zoom detection. We have tested the correctness by manual observation. If observed camera motion is similar to the detected one we say it correct, otherwise wrong. The Table- 1 contains the frame number for start-frame and the end-frame of each shot and result of pan and zoom detection. The sub-sampled cricket shots shown in this paper are taken from this cricket sequence only, e.g. Fig. 10 is the sub-sampled shot number 5(403-548). We observe that our scheme has been successful in correct classification in almost all the shots.

We have also tested our scheme on a number of other sequences as shown in Table- 2, and obtained about 90% correct classification. These results show that our scheme is capable of correct qualitative categorization of camera motion with real life video sequences. Compared to [6] which was tested only with video sequences captured in laboratory conditions, experimental results establish that our method is more general and widely applicable.

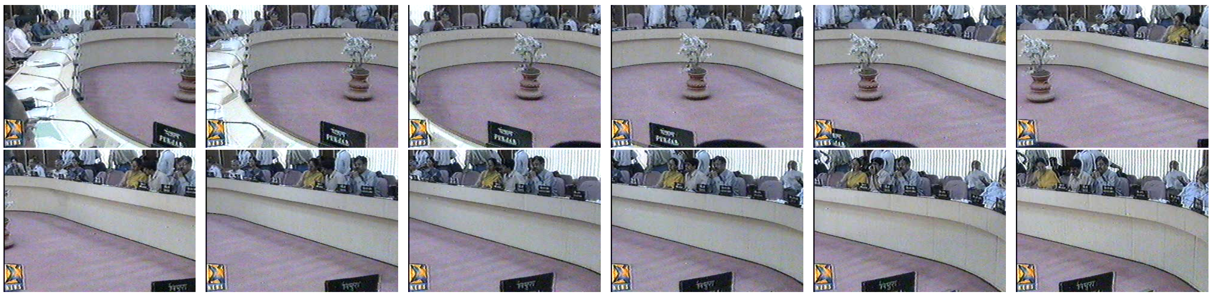
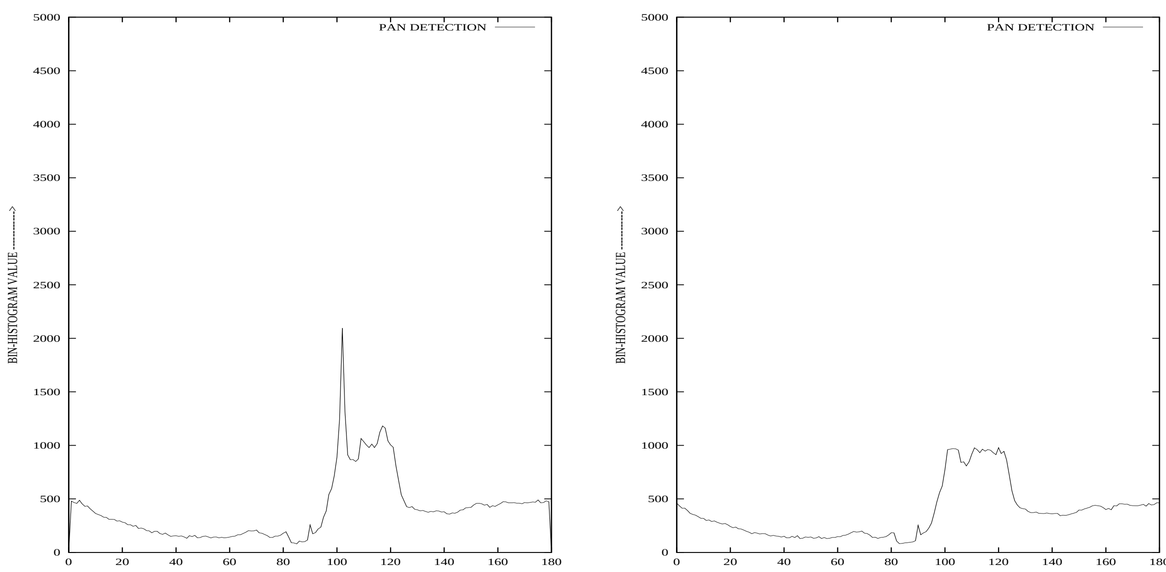


Figure 7: A Sub Sampled News Shot(Fast Panning)



(a) (b)

Figure 8: Phase Histogram:(a) Fast Panning (b) Slow Panning

Figure 9: A Sub Sampled Cricket Shot(Slow Panning)

Figure 10: A Sub Sampled Cricket Shot(Zoom-In)

5 Conclusions

In the present work we have presented a fuzzy theoretic approach for camera motion detection in video sequences. The motion vectors are primarily used as the key feature. Panning motion is detected by fuzzifying the angular value of optic flow vectors, while zooming motion is detected by fuzzifying the divergence of motion vectors.

6 Acknowledgement

We would like to thank Swati Jain for help.

SNO	SATRT	END	PAN	ZOOM
1	1	33	NO(correct)	NO(correct)
2	34	248	YES(correct)	NO(correct)
3	250	349	NO(correct)	NO(correct)
4	350	402	NO(correct)	NO(correct)
5	403	548	YES(wrong)	YES(correct)
6	549	557	NO(correct)	YES(correct)
7	558	570	YES(correct)	YES(correct)
8	572	655	YES(correct)	NO(correct)
9	656	688	NO(correct)	YES(wrong)
10	690	756	NO(correct)	NO(correct)
11	757	836	NO(correct)	NO(correct)
12	837	850	NO(correct)	NO(correct)

Table 1: Results on a Cricket Sequence

Sequence Name	Total No of Shots	No. of Shots Where Zoom Correctly Classified	No. Of Shots Where Zoom Incorrectly Classified	No. Of Shots Where Pan Correctly Classified	No. of Shots Where Pan Incorrectly Classified
Cricket	12	11	1	10	2
Classroom	20	19	1	19	Nil
Wildlife	14	14	Nil	14	Nil
Zeenews	16	16	Nil	16	Nil
Terminator	33	31	02	32	01
Colgate	24	23	01	22	02
Speed2	93	90	03	91	02

Table 2: Results: Camera Motion Detection

References

[1] A. Akutsu, Y. Tonomura, H. Hashimoto, and Yuji Ohba. Video indexing using motion vectors. In *Proceedings Visual Communication and Image Processing, SPIE, Amsterdam*, volume 1818, pages 1522–1530, 1992.

[2] Edoardo Ardizzone and M. La Cascia. Video indexing using optical flow fields. In *Proc. of IEEE Int. Conf. on Image Processing*, September 1996.

[3] N. Hirzalla and A. Karmouch. Detecting Cuts by Understanding Camera Operation for Video Indexing. *Journal of Visual Language and Computing*, 6:385–404, 1995.

[4] B.K.P. Horn and B.G. Shunk. *Robot Vision.* MIT PRESS, 1986.

[5] Alok Mittal, Aditya Valiaya, Subhashis Banerjee, and M. Balakrishnan. Real time vision systems for collision detection. *Journal of Computer Science and Informatics, Special Issue on Robotics and Automation*, 25(1):174–208, March 1995.

[6] M. V. Srinivasan, S. Venkatesh, and R. Hosie. Quantitative estimation of camera motion parameters from video sequences. *Pattern Recognition*, 30:593–606, 1997.

[7] G. Sudhir and J.C.M. Lee. Video annotation by motion interpretation using optical flow streams. *Journal of Visual Comm. Image Representation*, 7:354–368, 1997.

[8] Y.T. Tse and R.L. Baker. Global zoom/pan estimation and compensation for video compression. In *Proceedings ICASSP'91*, volume 4, May 1991.

[9] H.J. Zhang, A. Kankanhalli, and S.W. Smoliar. Automatic Partitioning of Full-motion Video. *Multimedia Systems*, 1(1):10–28, 1993.

[10] H.J. Zhang, C.Y. Low, and S.W. Smoliar. Video Indexing and Browsing using Compressed Data. *Multimedia Tools and Applications*, 1:91–113, 1995.