Fast Frame-Based Scene Change Detection in the Compressed Domain for MPEG-4 Video

Jens Brandt, Jens Trotzky, Lars Wolf

IBR
Technische Universität Braunschweig
Germany

Future Multimedia Networking - FMN'08
2008 September 17-18
Cardiff

Introduction

Scope

- Video adaptation for mobile devices
- Compressed domain video transcoding

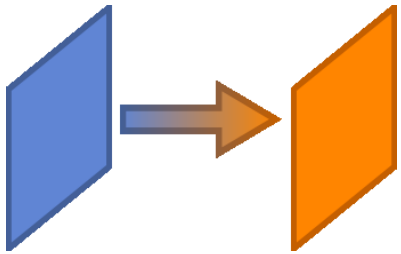
Problem

- How to determine transcoding parameters automatically?

Idea

- Analyse video content in the compressed domain
- Detect scene changes & special movements

Typical Approach



- Compute differences between successive frames
  - Computed in the pixel domain
  - Based on different mathematical models
  - Find edges
  - Use information about motion
- Detect scene changes based on the computed differences

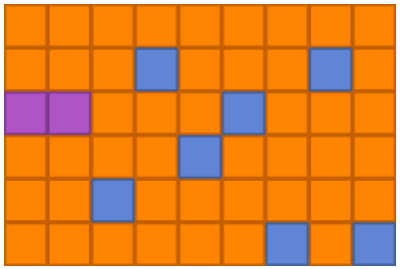
Our Approach

- MPEG compressed video already contains such differences
- Encoded in form of motion vectors and macro block types

$\Rightarrow$ Use these differences to detect scene changes

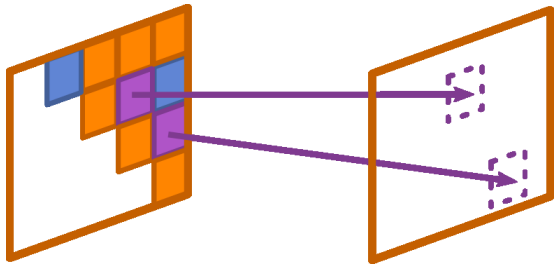
$\Rightarrow$ Here we concentrate solely on P-frames

MPEG-4 Video (Advanced Simple Profile)



- Frames are divided into 8×8 pixel blocks
- 64 DCT values per block
- Every set of four blocks builds one macro block (MB)
- Three different types of macro blocks:
  - inter-coded (with motion information)
  - intra-coded (without motion information)
  - not coded
- Motion vectors (MV) to encode motion between frames
- One or four vectors per inter-coded MB
- Most MV have the same direction as the motion

Compressed Domain Frame Analysis



Block Based Frame Measures

- Complexity = ratio of non-zero DCT values: $c = \frac{n_{\neg 0}}{64 \cdot n_B}$
- Intra-Ratio = ratio of intra-coded MB: $r_{INTRA} = \frac{n_{MB,INTRA}}{n_{MB}}$

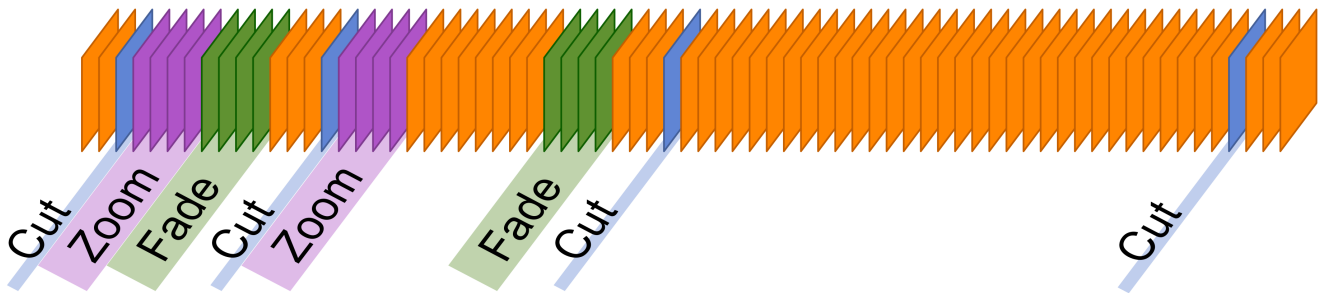
  $\Rightarrow$ this may be a hint about unsuccessful motion estimation

MV Based Frame Measure

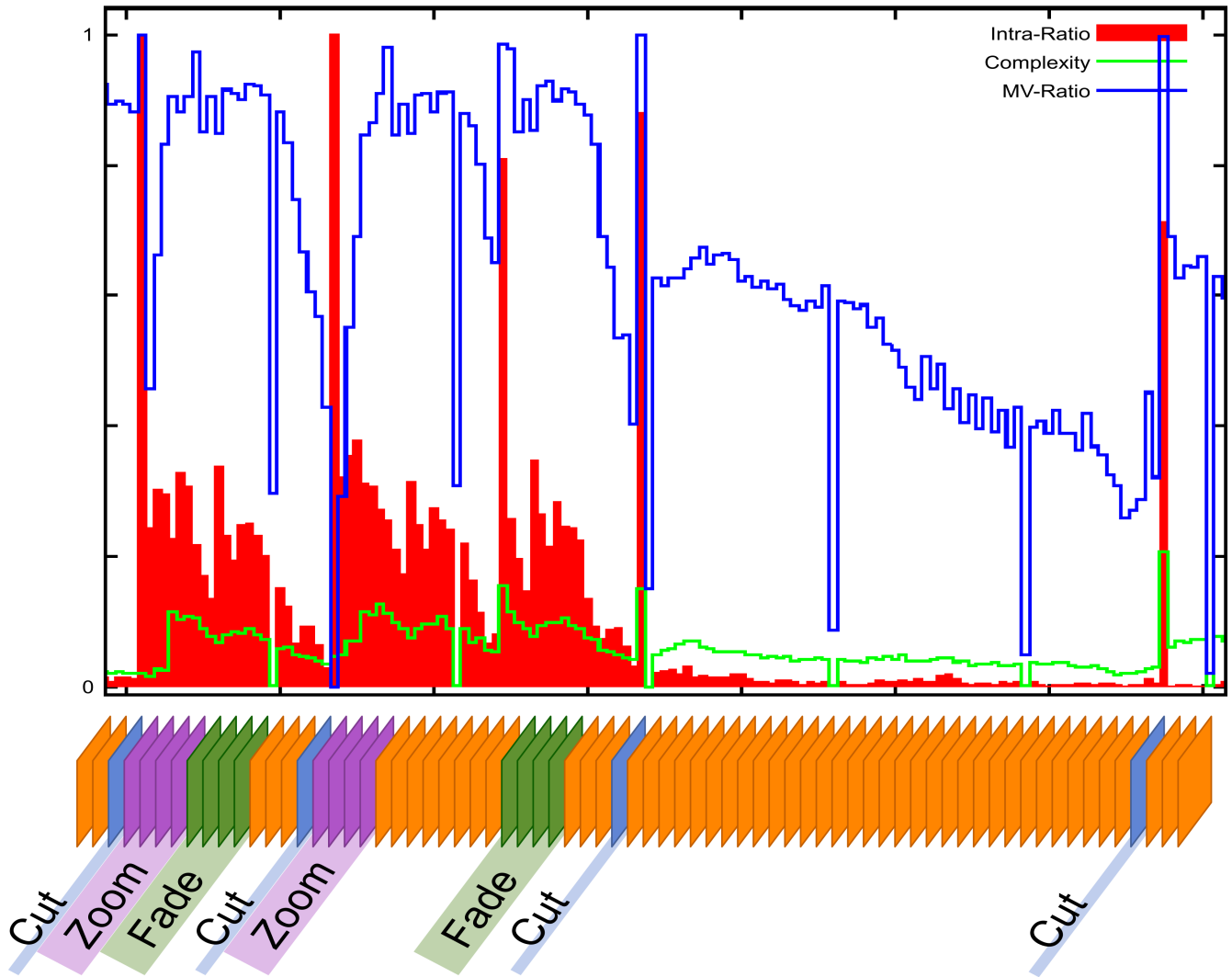
- Motion Vector Ratio = ratio of non-zero MV: $r_{MV} = \frac{n_{MV,\neg 0}}{n_{MAX,MV}}$

  $\Rightarrow$ measure for the amount of motion in the frame

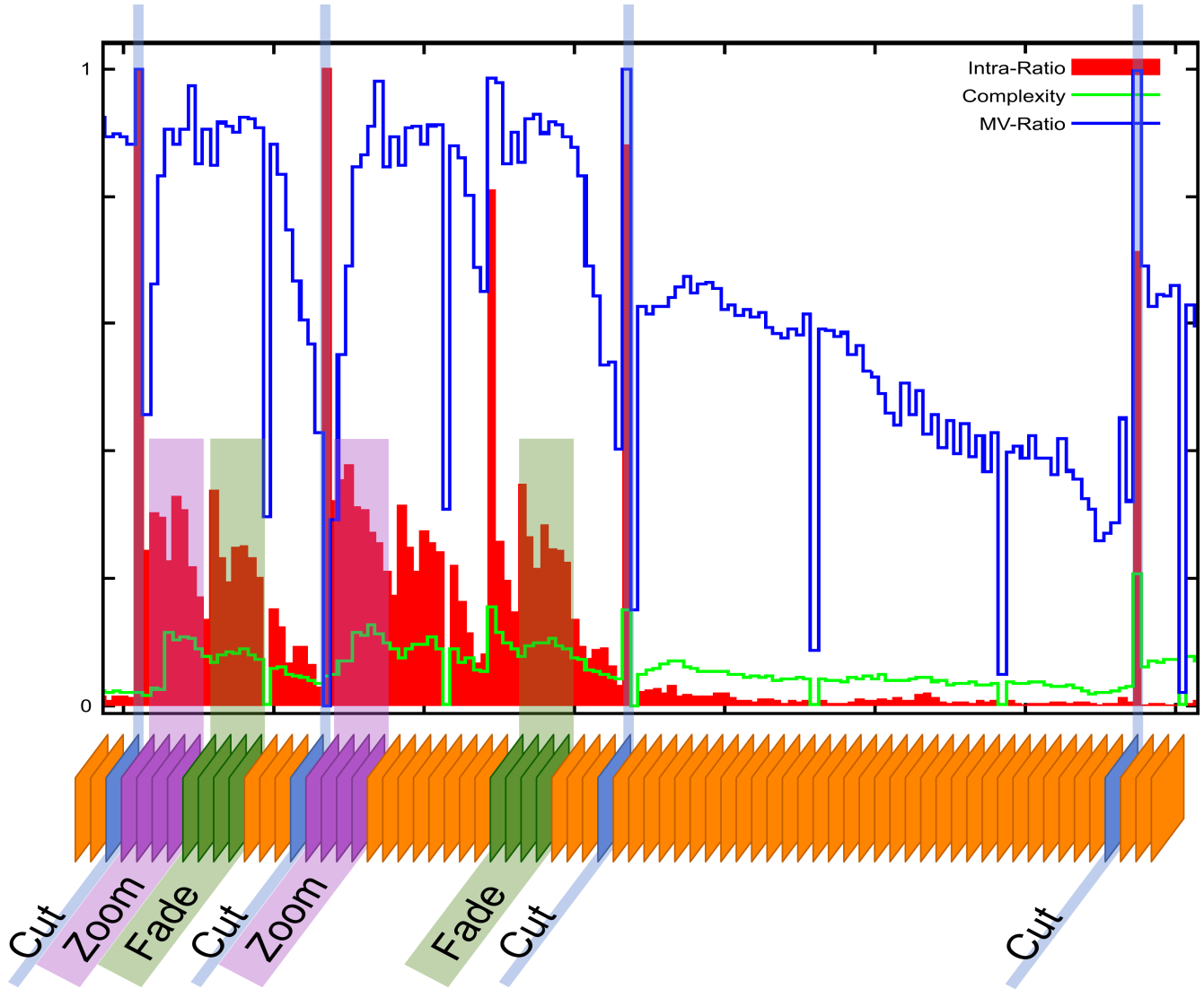
Example (Movie Trailer)



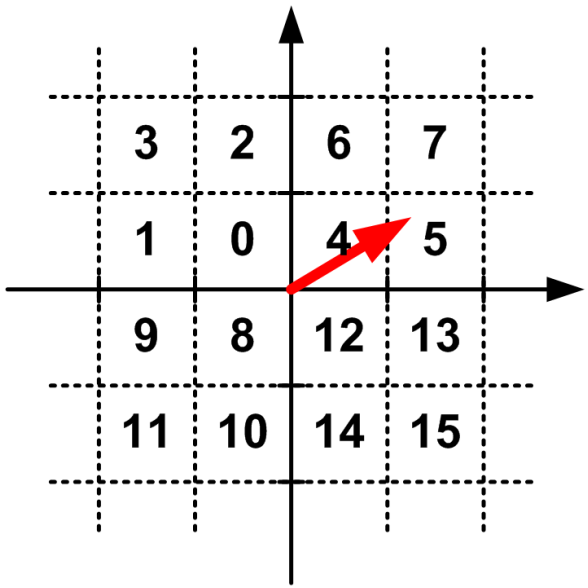
Example (Movie Trailer)



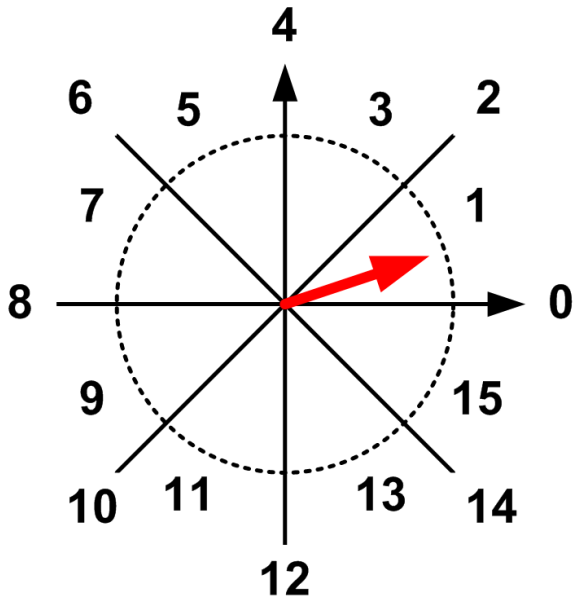
Example (Movie Trailer)



Motion Vector Histograms

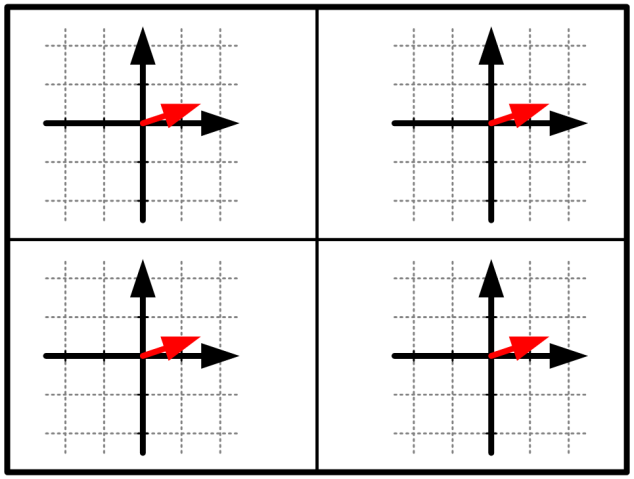


- Cartesian: classes correspond to length and direction



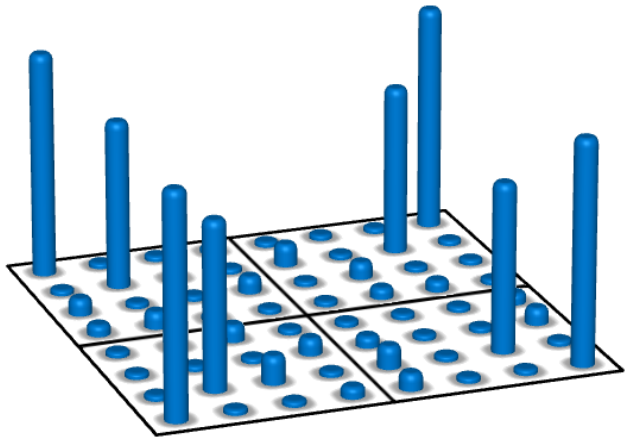
- Polar coordinate: polar sectors correspond to the direction

Frame Histogram

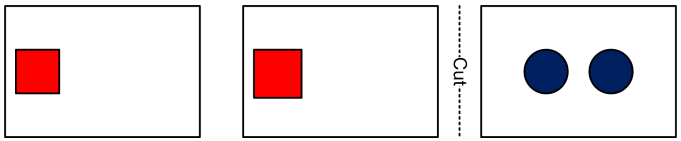
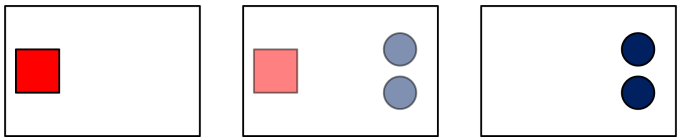
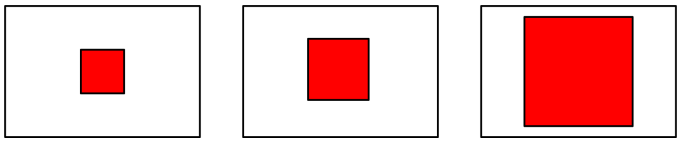


- Four quadrants per frame
- One histogram per quadrant
- Includes the origin of the MVs

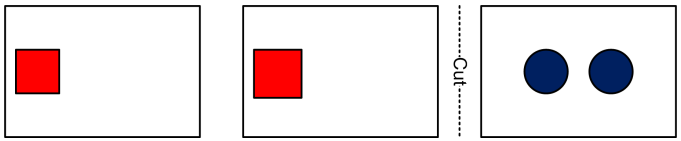
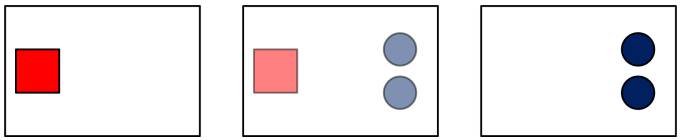
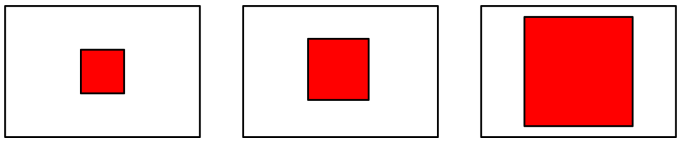
- Frame histogram of a zoom
- Most MVs are pointing outwards



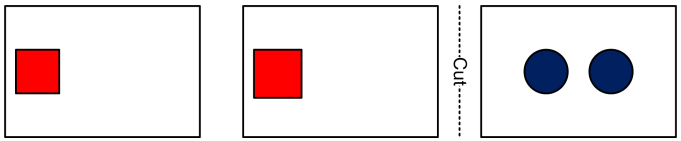
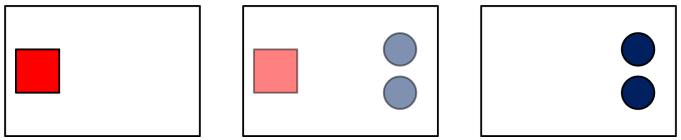
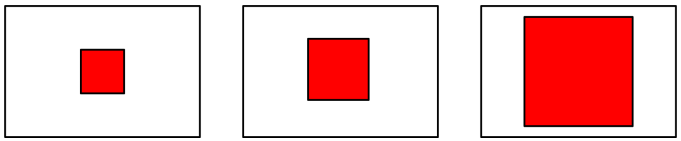
Scene Change Characteristics

	Complexity	Intra-Ratio	MV-Ratio
Cut 			
Fade 			
Zoom 			

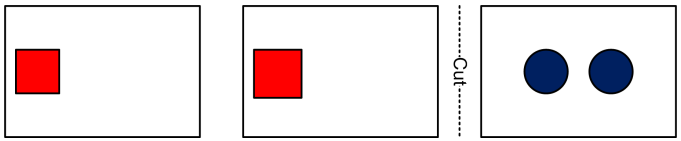
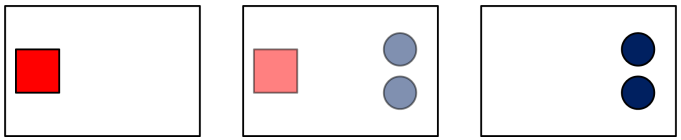
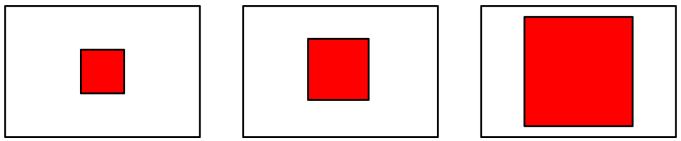
Scene Change Characteristics

	Complexity	Intra-Ratio	MV-Ratio
Cut 	high mostly > 0.1 also 0.04	very high mostly > 0.5 $> avg$	very high mostly > 0.95 $> avg$
Fade 			
Zoom 			

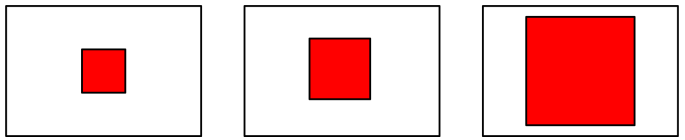
Scene Change Characteristics

	Complexity	Intra-Ratio	MV-Ratio
Cut 	high mostly > 0.1 also 0.04	very high mostly > 0.5 $> avg$	very high mostly > 0.95 $> avg$
Fade 	medium	medium mostly > 0.15 $> avg$	high mostly > 0.8
Zoom 			

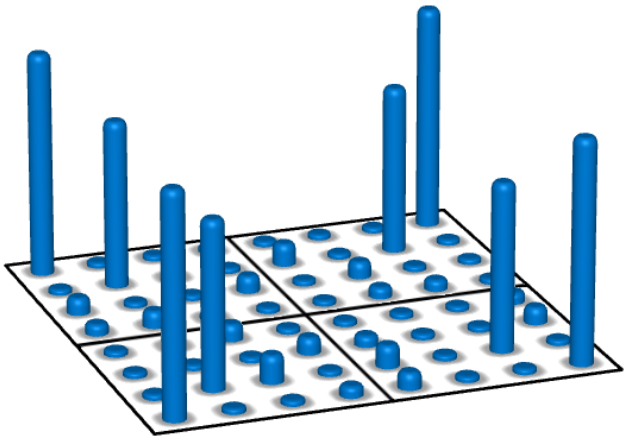
Scene Change Characteristics

	Complexity	Intra-Ratio	MV-Ratio
Cut 	high mostly > 0.1 also 0.04	very high mostly > 0.5 $> avg$	very high mostly > 0.95 $> avg$
Fade 	medium	medium mostly > 0.15 $> avg$	high mostly > 0.8
Zoom 	similar to	the situation	of fades

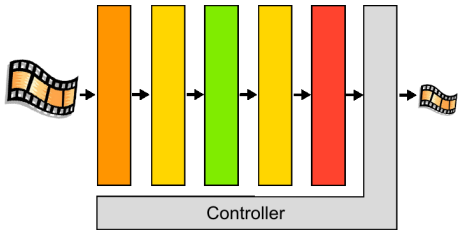
Zoom Detection



- Complexity, Intra-Ratio and MV Ratio similar to fades
- MVs are pointing inwards or outwards
- Use histograms to count such MVs
- A Zoom is detected if
  - Number of zoom indicative vectors is 30% higher than expected for both types of histograms, or
  - Number of zoom indicative vectors is 100% higher than expected for one type of histograms



Implementation & Evaluation



Implementation

- Integrated into our transcoder implementation
- Transcoding module which analyses each frame

Evaluation

- 2 movie trailers, 1 news sequence and 1 soccer game sequence
- Duration of 90 seconds
- Resolution between 320×240 and 1280×720 pixels
- Encoded with modified FFmpeg (only P-frames)

Evaluation

Reference

- Manual frame by frame analysis
- Cuts consists of exactly one frame
- Fades and zooms last at least two frames

Results

Video	Cut	Fade	Zoom
	existing/detected/false positives		
news	14/13/2	1/1/2	2/2/0
soccer	4/4/1	5/5/3	9/7/0
movie-1	51/48/1	8/7/4	3/3/0
movie-2	49/38/11	14/13/2	7/7/0

Evaluation (cont.)

- Most falsely detected frames belong to another type of scene change:
  - Falsely detected cuts belong to fast fades
  - Undetected cuts are detected as single frame fades
  - Moving background is detected as a fade
  - Changing background color is detected as a cut
- Average processing time per frame:
  - 0.6 to 6.5 ms
  - 18% to 33% of time needed for bit stream parsing

  $\Rightarrow$ Very low overhead

Conclusion

- Frame-based compressed domain scene change detection
- Analysis of DCT values and motion vectors of MPEG-4 video
- Three different measures
- Motion vector histograms
- Easy and fast computation
- Promising evaluation results

Lessons Learned

- Detecting the type of scene change can be hard
- Some situations are challenging even for humans
  - Abruptly changing light conditions (e.g. flashlights)
  - Many distributed fine movements (e.g. bubbles in water)
  - Close-ups with moving background

Questions?

Jens Brandt, Jens Trotzky, Lars Wolf
{brandt|trotzky|wolf}@ibr.cs.tu-bs.de

IBR
Technische Universität Braunschweig
Germany

http://www.ibr.cs.tu-bs.de