# GPU-accelerated detection of severe video distortions

**Hannes Fassold**, Jakub Rosner, Martin Winter, Peter Schallauer

JOANNEUM RESEARCH
Forschungsgesellschaft mbH

Institute for Information and Communication Technologies

Steyrergasse 17
8010 Graz, Austria

Tel. +43 316 876-50 00
Fax +43 316 876-50 10

hannes.fassold@joanneum.at
www.joanneum.at/digital

## Introduction

- Millions of hours of analog video material waiting for digitization in archives of broadcaster
- Automatic and high speed content-based quality assessment of the material would be very valuable
- Here we focus on severe analog video distortions ('video breakup')
  - Often affect whole line at once
  - Line jittering, color shifting, dropouts

## Algorithm

- Based on (optionally motion-compensated) difference image of consecutive frames
- Two different measures (row change, edge ratio) on the difference image are used for frame-wise decision
- Some temporal post-processing is done

## Row change measure

- Algorithm
  - Reduce difference image to a vector by calculating a statistical measure (e.g. mean, median) for each row in difference image
  - Row change measure is distance between respective vectors from consecutive frames
- Fermi GPU implementation - key issues
  - Difficult to implement the row histogram efficiently with atomic functions because row histogram is highly concentrated at 'zero'
  - Solved by employing **multiple** 8-bit histograms in shared memory during row histogram generation (with atomic functions) and merging them laterly
  - 48 KB shared memory (per SM) provided by Fermi GPUs are crucially needed

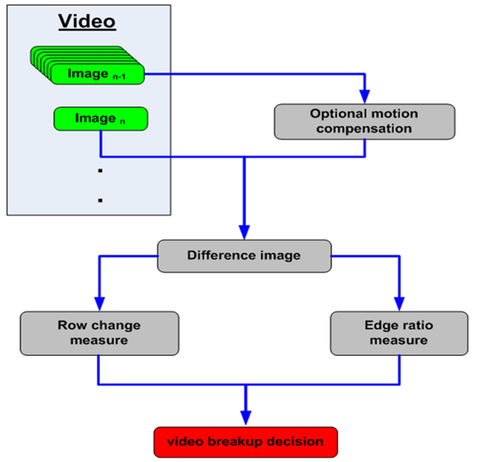


Figure 1: Schematic workflow of the algorithm


Figure 2: Appearances of severe analog video distorations

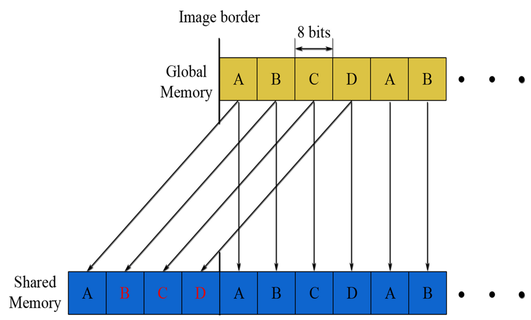


Figure 3: Loading 8-bit elements for image apron from global to shared memory

## Edge ratio measure

- Algorithm
  - Based on the ratio between horizontal and vertical edges in the difference image
- Fermi GPU implementation - key issues
  - Each threads handles four consecutive pixels of data type **uint8**
  - Efficient loading of 1-pixel 'apron' for convolution kernel (see figure 3) for image border handling

## Evaluation

- Algorithm achieves recall of 82 % with 0.44 FP per minute video
- GPU implementation processes **ten** Full HD (1920 x 1080) video streams in real-time on a GTX 480

## Conclusion

- Algorithm achieves sufficient detection quality for multiple Full HD streams in real-time
- Transfer time GPU-CPU dominates runtime of GPU implementation (see figure 4)
- Integrated in content-based quality analysis prototype application (http://www.av-inspector.com)

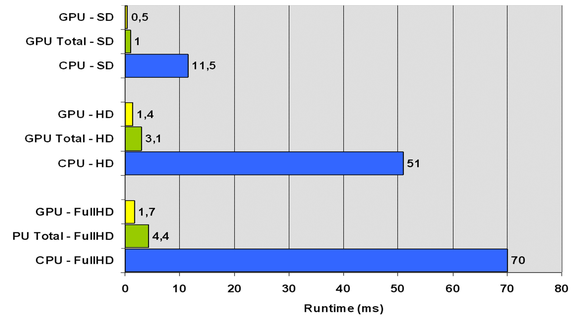


Figure 4: Runtime comparison of GPU and multi-threaded CPU implementation (GPU Total = GPU implementation inclusive transfer time CPU-GPU)

This work was supported by the European Commission under the contract FP7-231161"PrestoPRIME", www.prestoprime.org