BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY

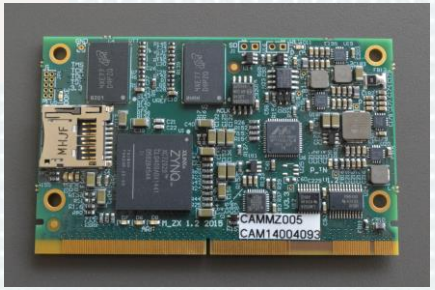
Embedded Video Processing for Smart Cameras

Pavel Zemčík, Martin Musil, Petr Musil, Svetozár Nosko

2018

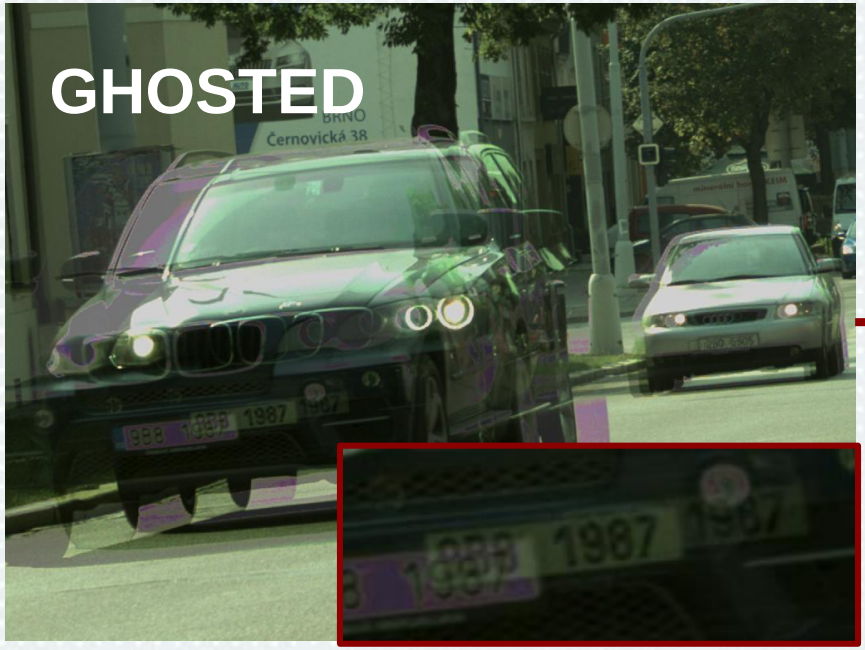
Motivation

- Video is becoming increasingly popular and available – in Industry 4.0 applications and also in many other ones (traffic, physics, etc.)
- Cameras produce huge amount of Information; do we want to transfer it all to computer and process/store it there?
- Probably not
  so we really need
  „intelligent cameras“



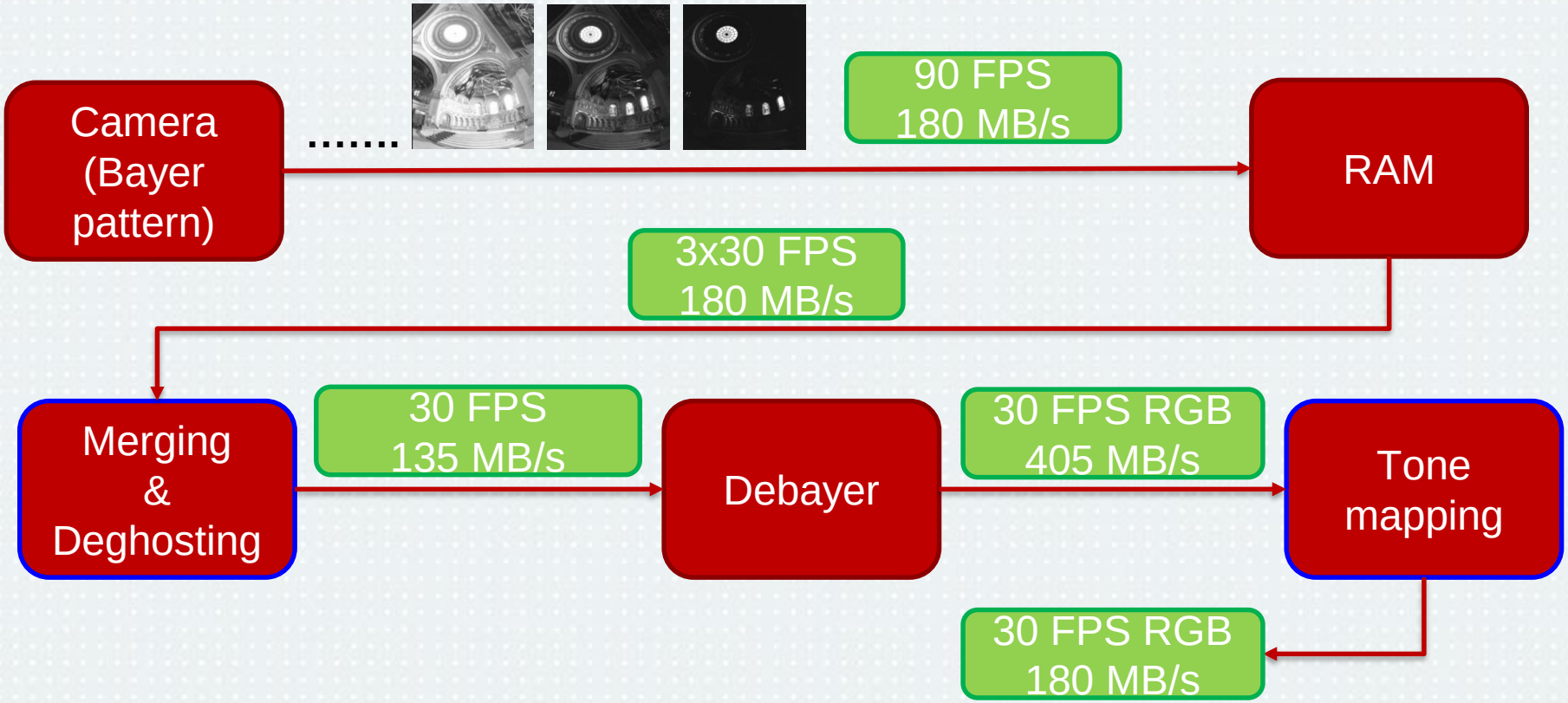
Application Example HDR

Example with Ghosts



HDR acquisition

- High throughput required - 180MPix/s (30FPS)
  - Everything running in parallel

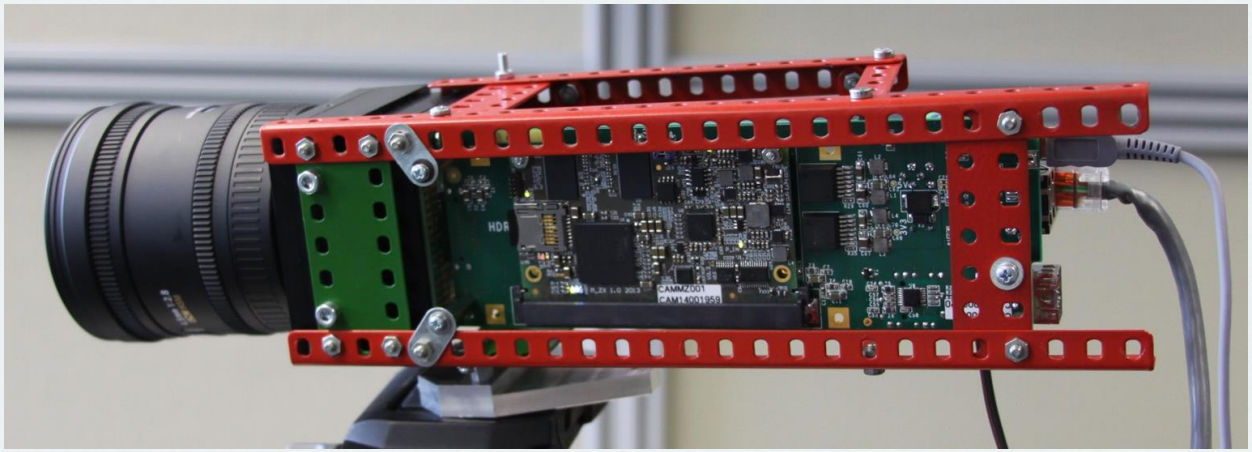


FPGA Specifics

- Implicitly parallel
  - suitable for high-throughput applications
- Programmable hardware structure
  - Custom & complex computing units
  - Distributed memory (Registers, BlockRAM)
- Described by HDL languages
  - Long development time
  - HLS becomes popular
- Low-power and high performance
- In some chips combined with CPU – nice!

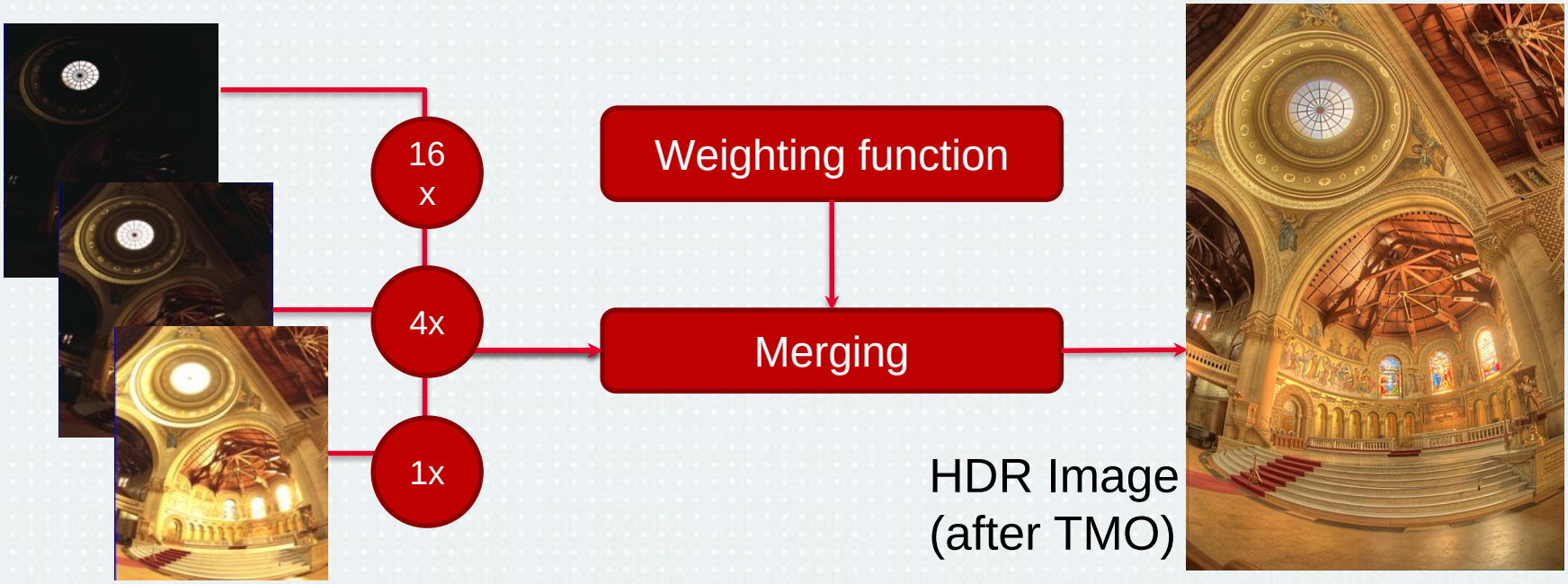
Custom Camera

- 2MPix color CMOS
  - Up to 240FPS@FullHD
  - Directly conn. to FPGA
- ARM + FPGA (Xilinx Zynq)
  - ARM running Linux
- Hardware H.264 encoder



HDR Acquisition

- 2-3 images merged into one HDR
- Debevec & Malik technique
- Varying exposure times: e.g. 1, 4 and 16ms



Tone Mapping

- Need to compress the HDR data into 8 bit RGB

Luminance

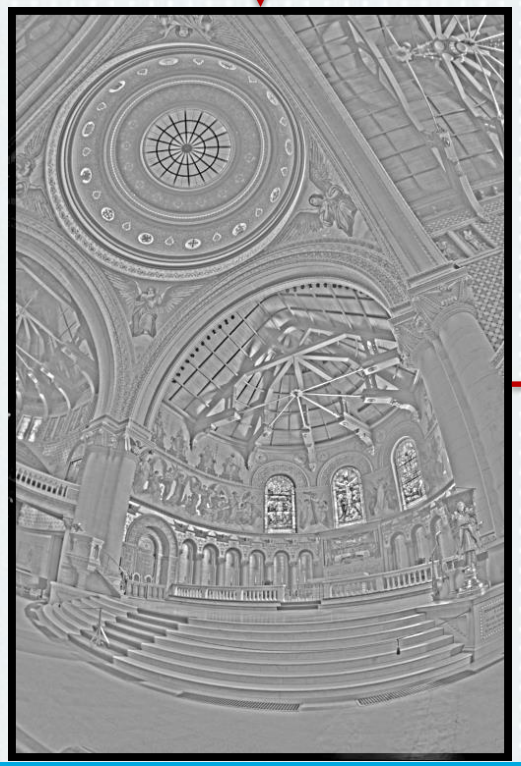
Bilateral filtering

1

Base layer

-

2



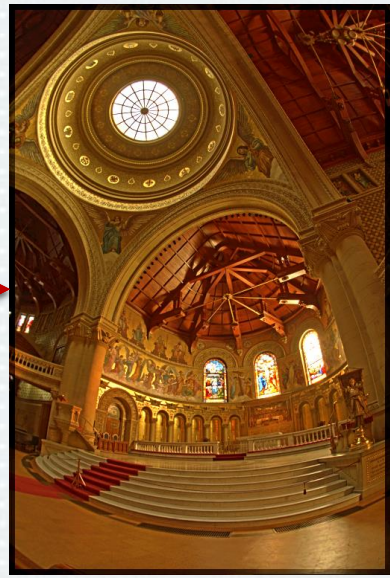
Detail layer

Look-up DR compression

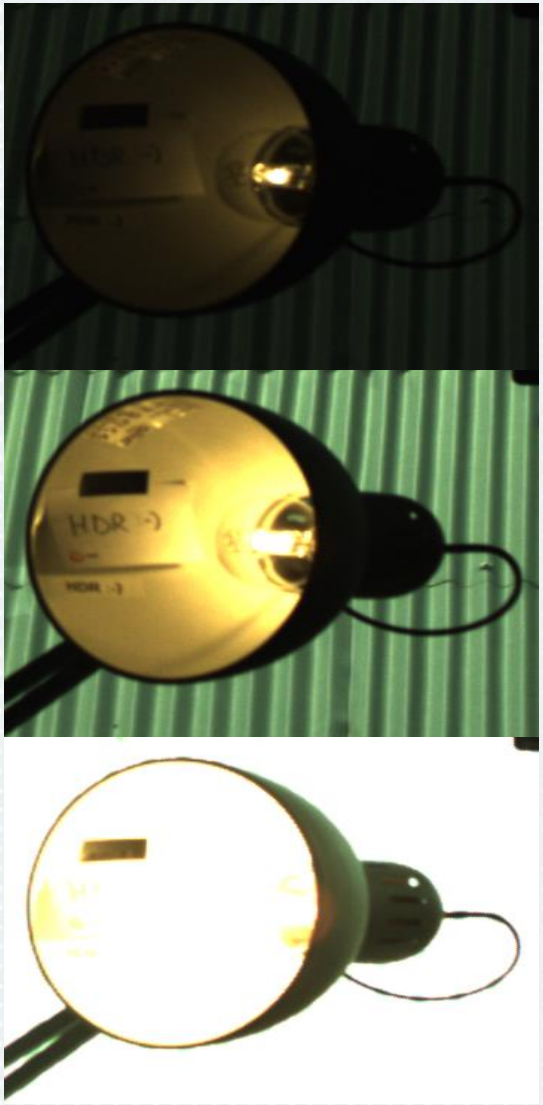
+

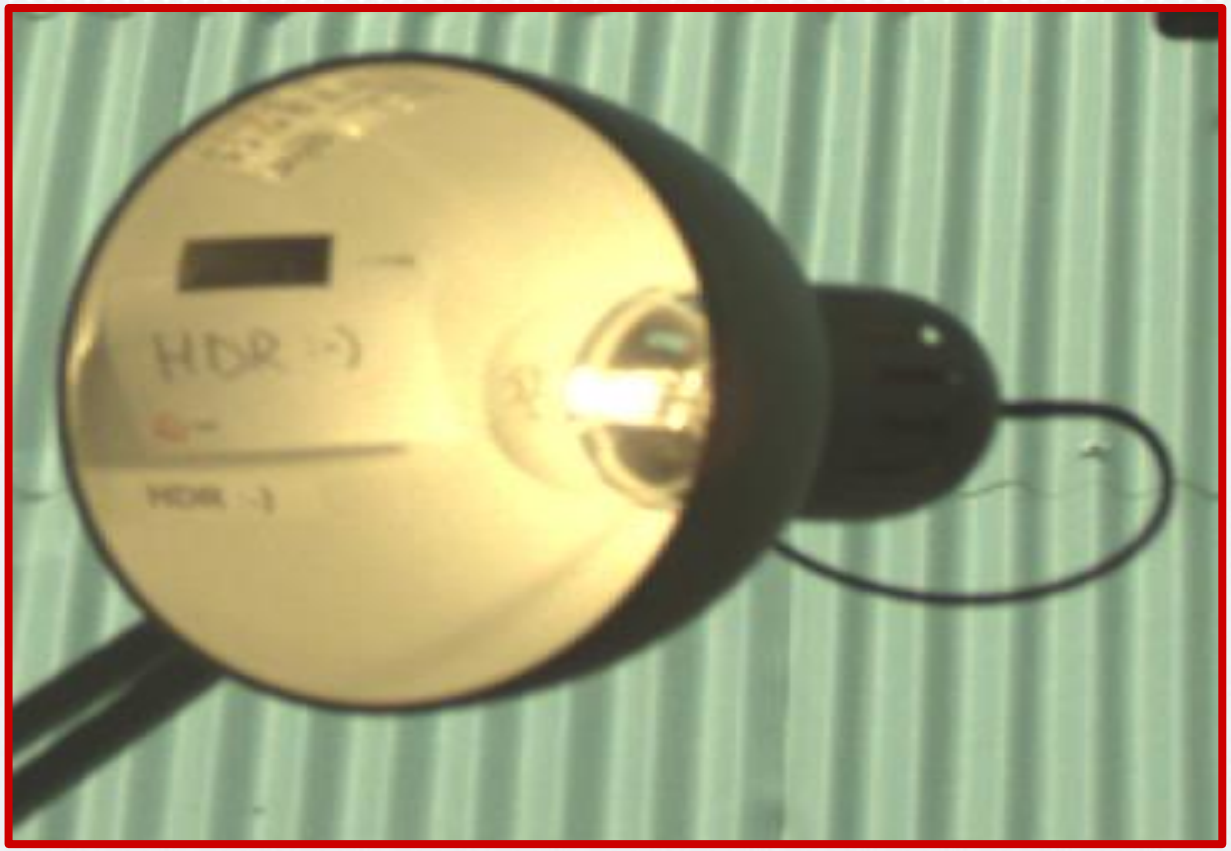
3

Result



Example HDR





HDR

HDR Ghosting

- Hold on! Stand still! Capturing the HDR image...
  - What if… something moves?
  - Sequential capture … „ghosts“
- Static camera assumed

Deghosting

- Good visual result = computing complexity
  - Offline computing, up to minutes per frame
  - Patch based & optical flow solution

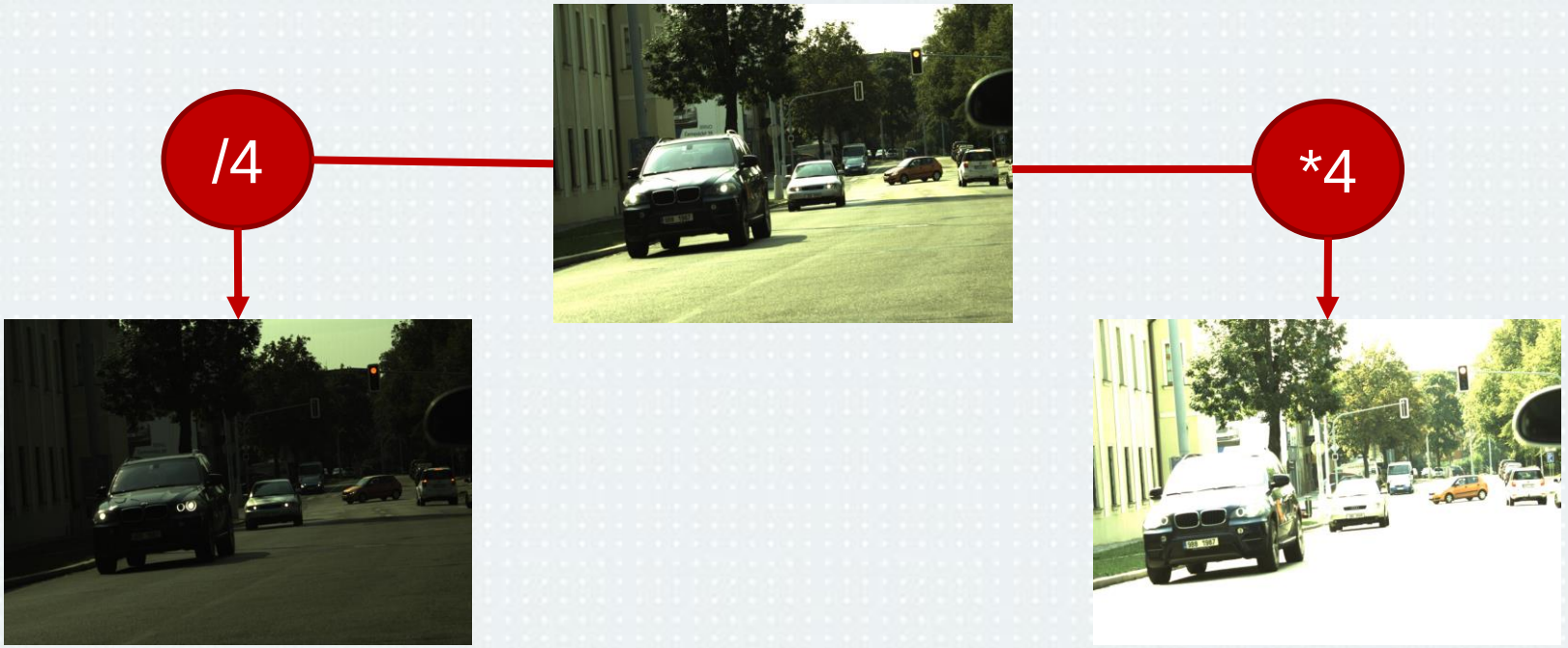
  … solutions for PC available but very „expensive“ …

- Real-time processing
  - Compromise between quality & performance
  - Cannot afford optical flow
  - Does not have to be “perfect”

  … but we want in in the embedded system …

Proposed algorithm

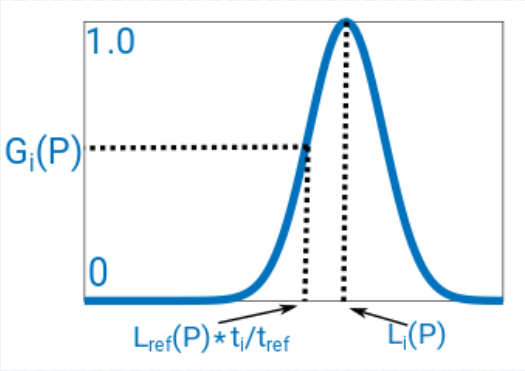
- Based on pixel value matching
  - Matching pixel value in image sequence
  - Known exposure time – does it „fit“?



Certainty maps

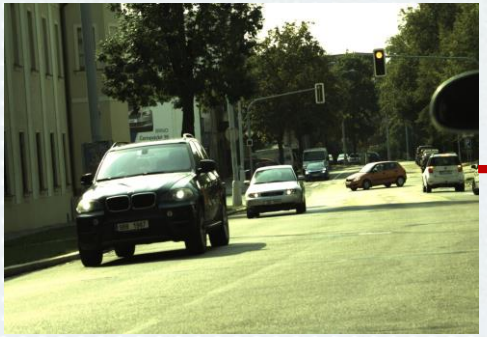
- Normal distrib. of pixel values V:

$$H = \sum_{i=1}^{n} (2 - C_i) * w(L_{ref}) * \frac{t_{ref}}{t_1} + C_i * w(L_i) * \frac{t_i}{t_1}$$



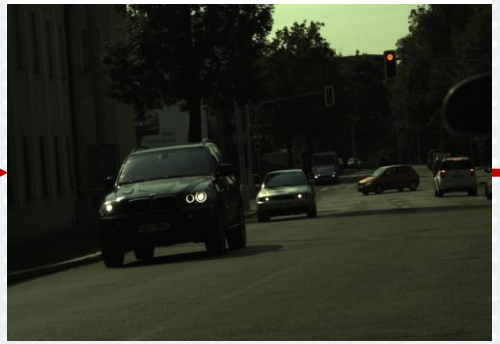
diff(Prediction,real) ---> Certainty maps

exp*4



Reference

exp/4




THE
BIGGEST
BEAR


THE BIGGEST BEAR
Adam Relf
SCHOLASTIC

Conclusions

- Embedded video processing feasible
- HDR and/or object detection implemented
- Inexpensive solution with high performance and very low power consumption
- Algorithms quite difficult to implement
- Anyhow, many very feasible applications exist
- With existing hardware, very special image acquisition is possible, perhaps even for CERN applications, definitely for Industry 4.0