
aspire invent achieve
imec

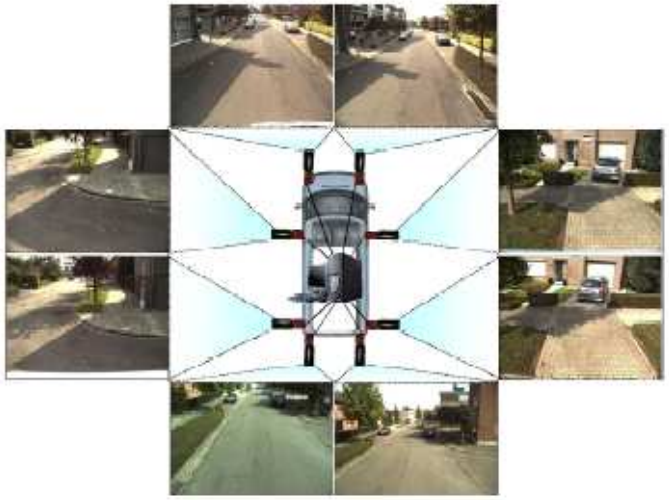
URBAN WP 1

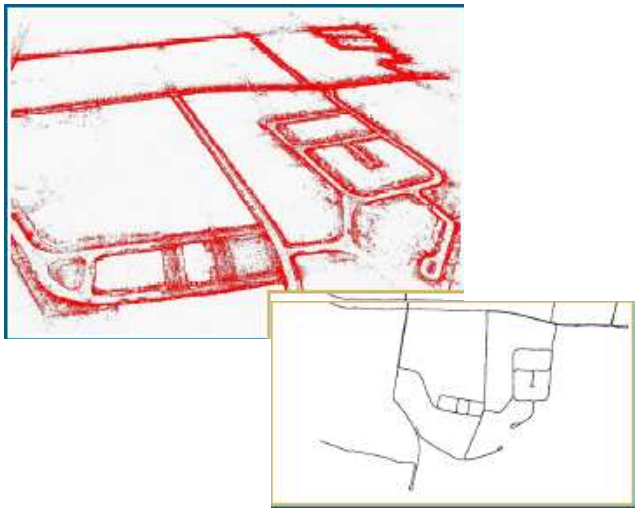
Towards real-time video de-bayering and compression: benchmarking and initial optimizations



Work package 1: Digital Surveying

- Digital Surveying
  - Processing on board of the van
    - Multi-video capture
      - de-bayering and coding
  - Processing on the servers
    - 3d reconstruction
      - decoding, automatic feature extraction and matching…



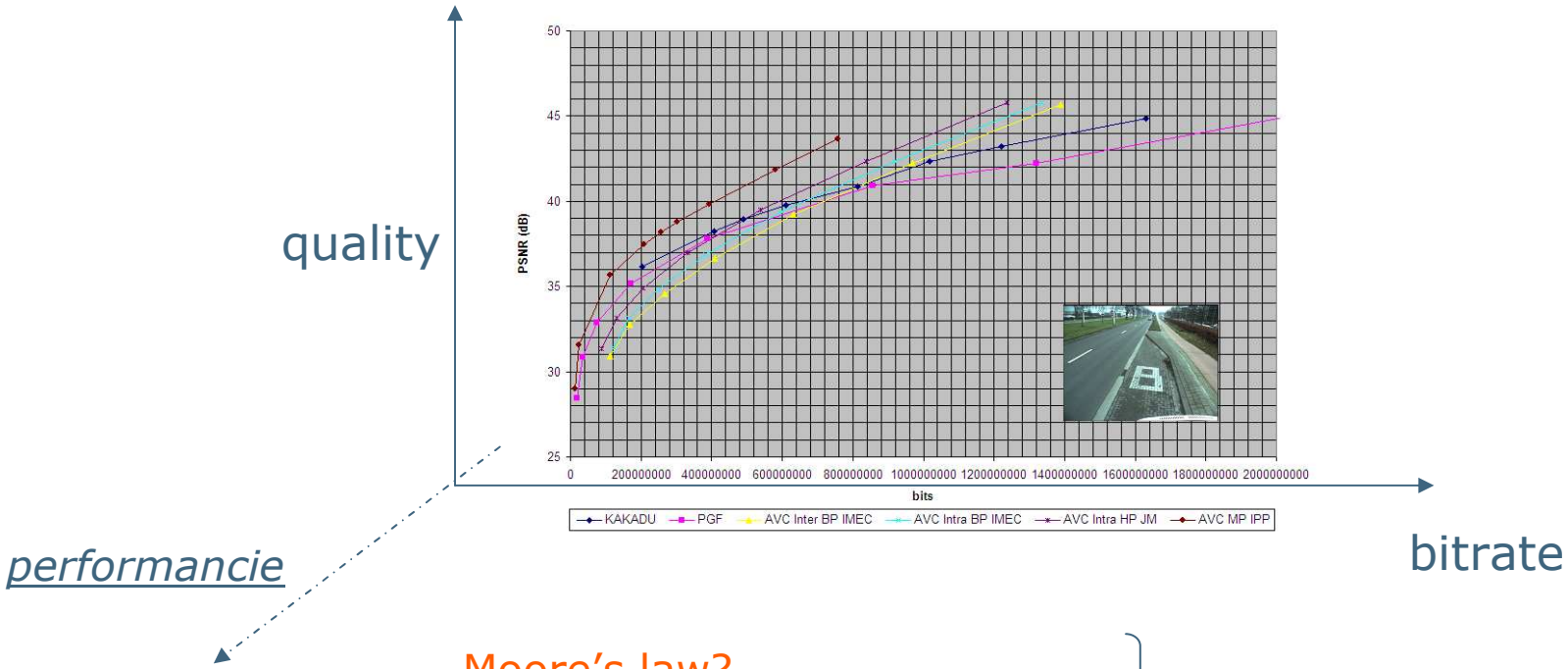


© imec restricted 2008

Work package 1: Digital Surveying

- Main Goal:

  Algorithm optimizations for lower computing complexity and improved efficiency



Moore's law?

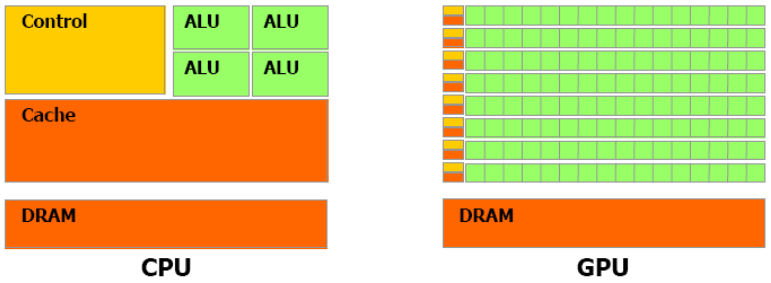
ILP, memory and power walls

Parallelization

© imec restricted 2008

Hardware and Parallelization

- Parallel programming
  - Not all algorithms are parallelizable (e.g. entropy coding…)
  - Modifications to enable parallelism at the expense of quality performances
  - Different algorithms call for different programming patterns for different strategies and hardware…
- CPU:
    - Multithread/Multicore: OpenMP
    - SIMD: MMX/SSE e.g. Intel's Performance Primitives lib. (IPP)
- GPU
  - General Purpose GPU programming
  - CUDA programming model



© imec restricted 2008

Task 1.1: On-board the recording van

Benchmark on video compression tools



D1.1.1 quality analysis and selection of compression tools to optimize

- Benchmark of state-of-the-art video compression algorithms:
  - Encoders:
    - DCT- and block-based algorithms: AVC (H.264)
      - AVC (H.264) : reference software JM 13.0, Intel IPP and IMEC encoders
    - DWT- based algorithms:
      - JPEG 2000 : Kakadu 5.2.4
      - PGF : pgfconsole 1.0
  - Data:
    - 8 x 200 frames at 12 fps in 1628 x 1236 format

© imec restricted 2008

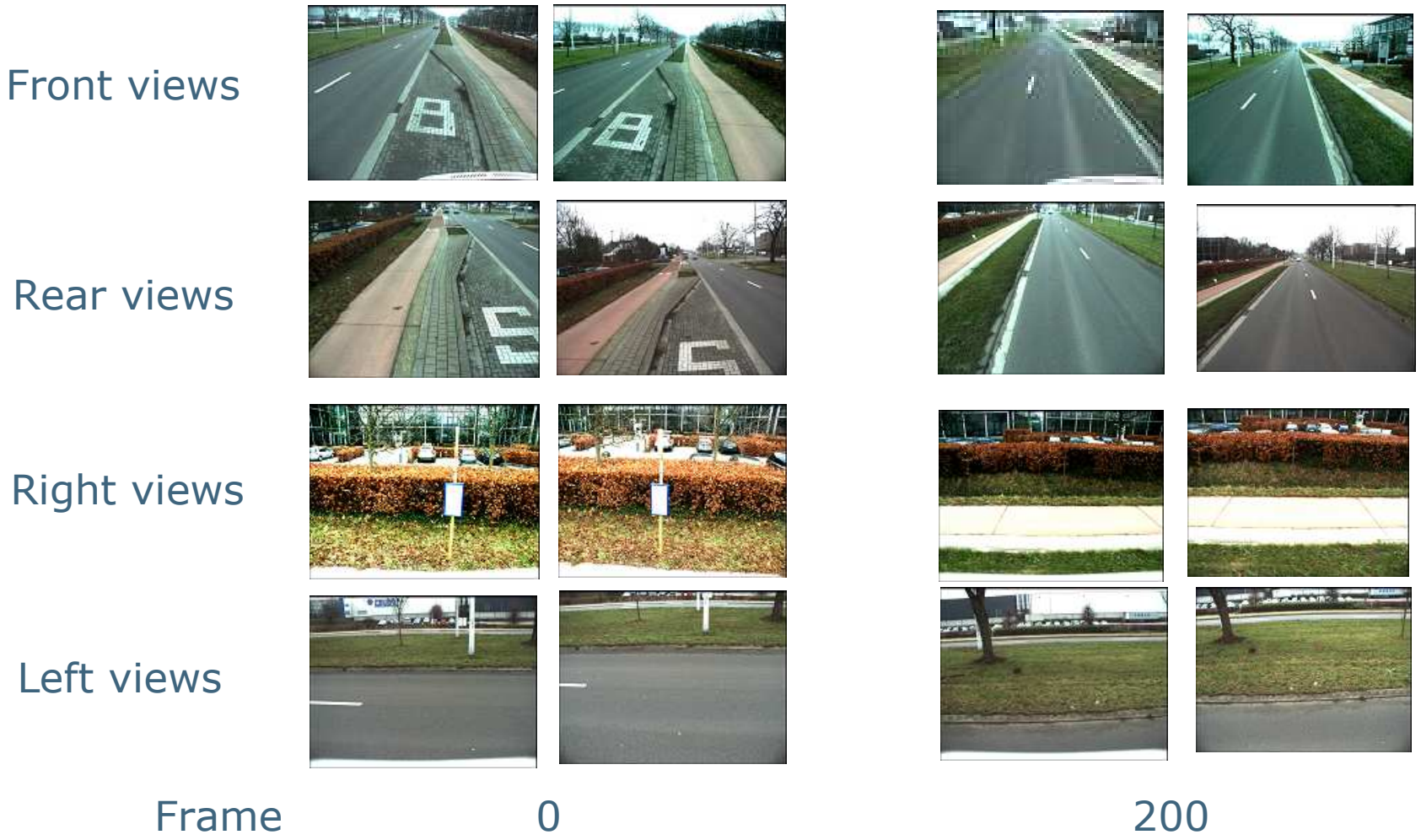
D1.1.1 quality analysis and selection of compression tools to optimize

- Related benchmarks:
  - *European Broadcasting Union (EBU) has organized objective and subjective quality experiments for different image/video HD resolutions between JPEG2000, H.264 (AVC) and MPEG-2. Results due to Feb. 2009*
  - Rate Distortion (objective metrics only)
    - Marpe et al. 2005:
      - low and intermediate resolutions: AVC Intra Main Profile, for HD: JPEG2000
    - Topiwala 2005 and JVT team of ITU-T 2004:
      - AVC HP Intra offers R-D gain around 0.2 and 1 dB in PSNR over JPEG2000
    - Ouaret et Ebrahimi 2006:
      - JPEG2000 is very competitive with AVC HP Intra
  - Objective and subjective assessment:
    - Cho et al. 2007 :
      - at high bitrates (high quality) AVC distortions are less perceived than for JPEG2000
      - At low bitrates, blocking effects of AVC are more annoying than distortions caused by wavelets

© imec restricted 2008

D1.1.1 quality analysis and selection of compression tools to optimize

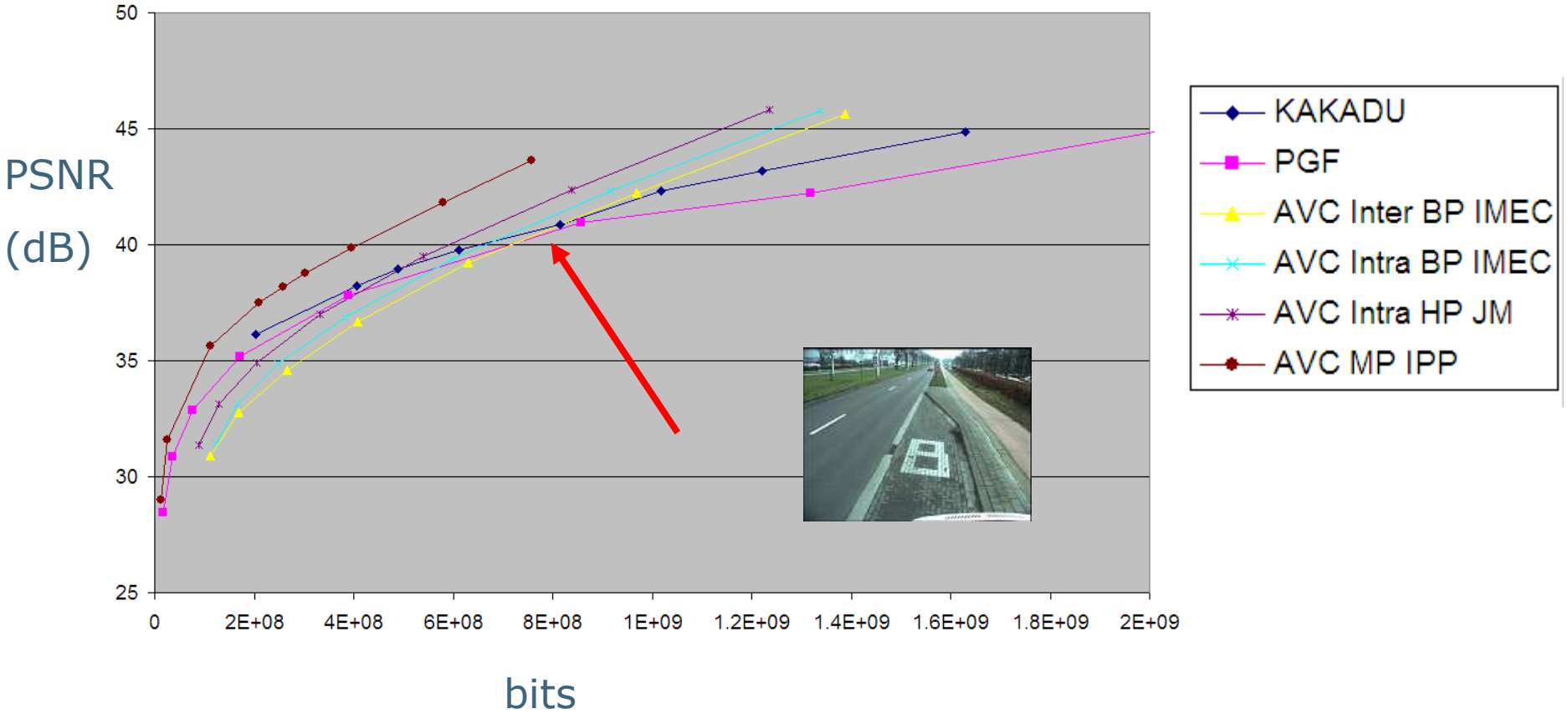
- Data



© imec restricted 2008

D1.1.1 quality analysis and selection of compression tools to optimize

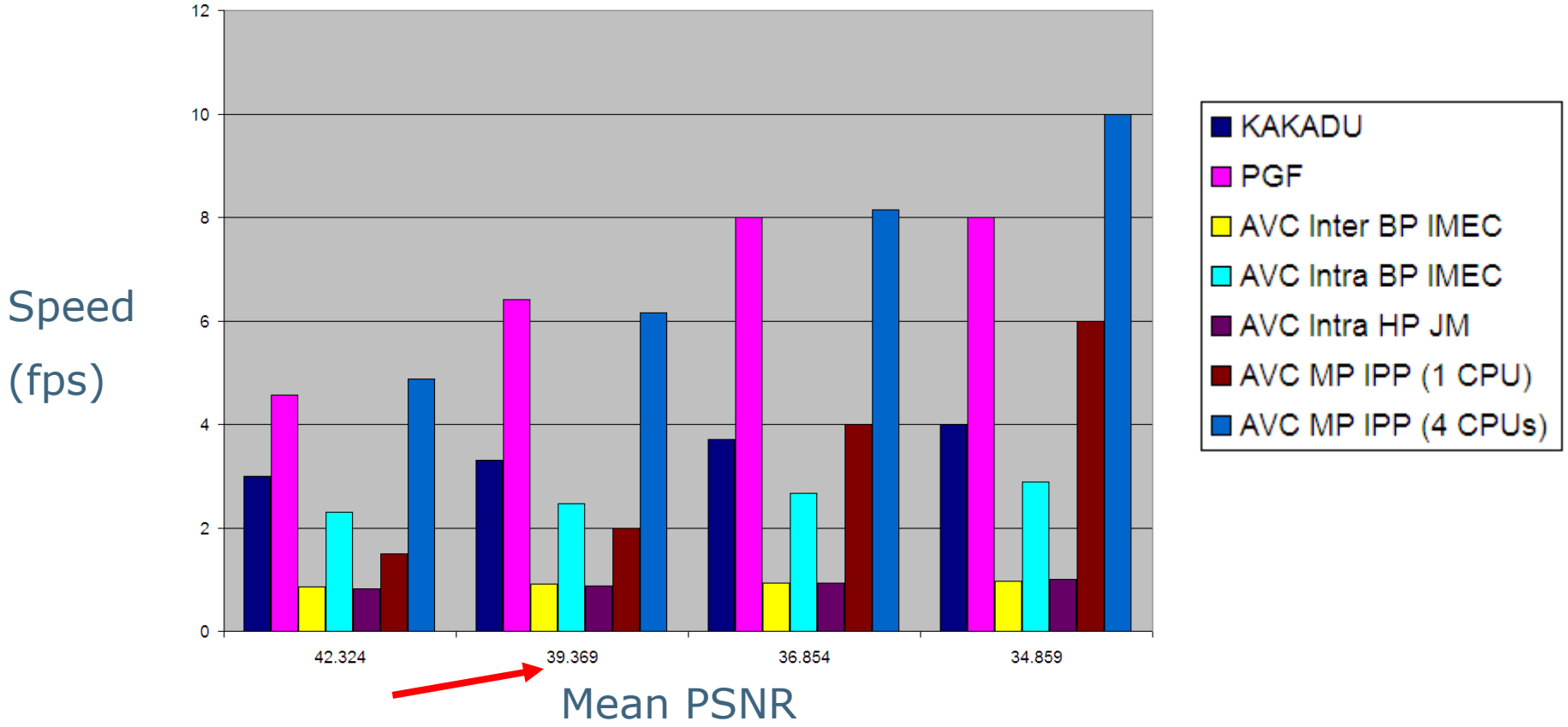
- Front and rear views
  - Quality measured with SNR on Y component
  - Total bits for 200 frames



© imec restricted 2008

D1.1.1 quality analysis and selection of compression tools to optimize

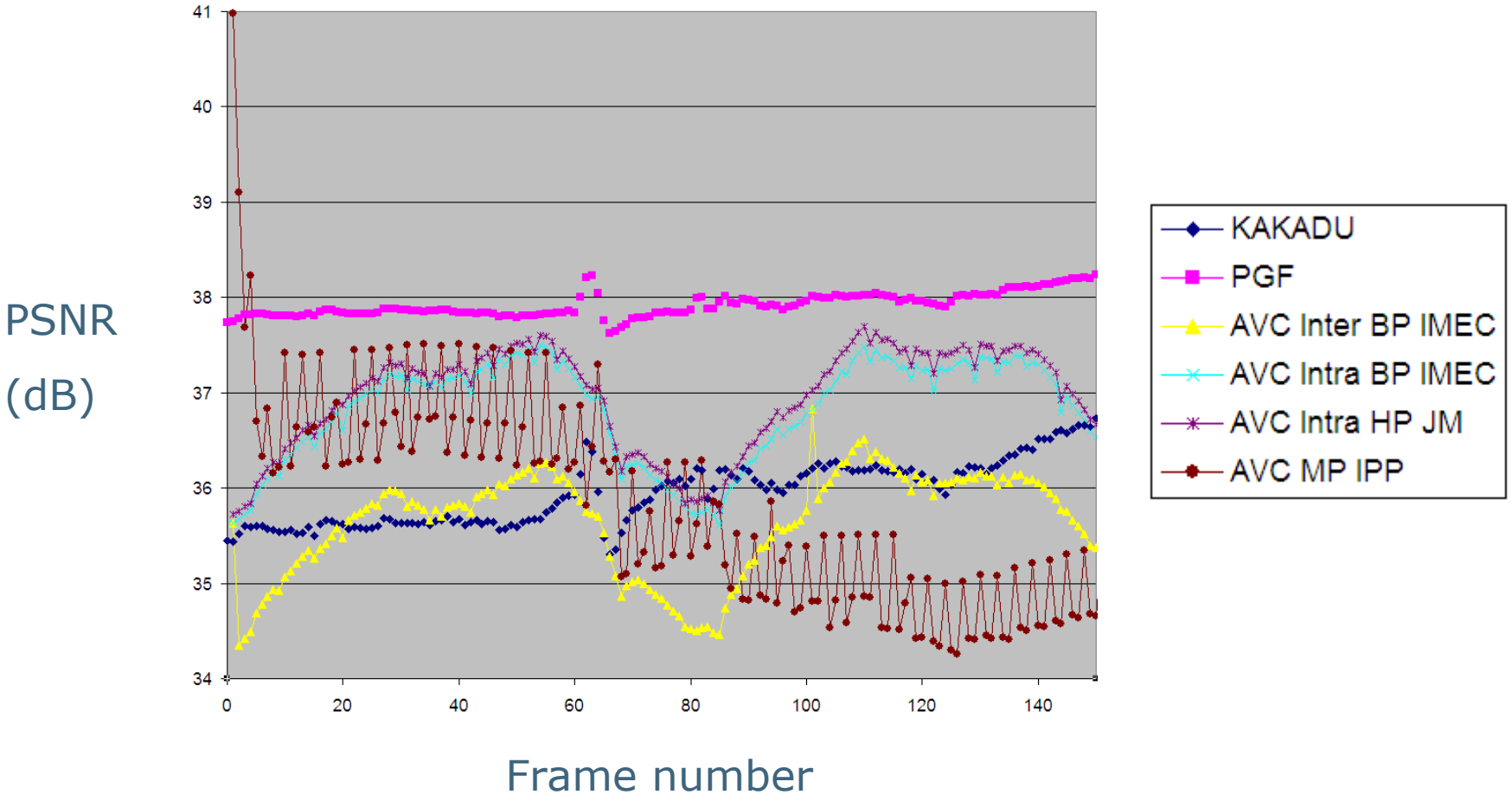
- Speed measurements
  - Intel Core 2 Quad CPU @ 2.4 GHz, 2 GB RAM
  - CPUs do not reach the real-time constraint at the desired quality



© imec restricted 2008

D1.1.1 quality analysis and selection of compression tools to optimize

- Temporal Analysis



© imec restricted 2008

D1.1.1 quality analysis and selection of compression tools to optimize

- Conclusions on the benchmark
  - Compression performances highly depend on the content:
    - Quality - Similar results with Cho et al 2007 (subjective tests):
      - For lower bitrates wavelets outperform AVC Baseline Profile and AVC Intra High Profile
      - Vice versa for higher bitrates
    - Speed/complexity:
      - Data dependency!
      - Kakadu, PGF and Intel's AVC are the fastest but do not reach 12 fps
      - Imec AVC Baseline Profile is as fast or faster than Intel's AVC when disabling parallelism
    - Temporal stability:
      - AVC shows significant variations between frames when compared to Wavelets
  - Parallelization of AVC Intra or Intra High Profile on CPU (IPP / Hyperthreading with OpenMP) or GPU (CUDA)?
  - More info available in deliverable D 1.1.1

© imec restricted 2008

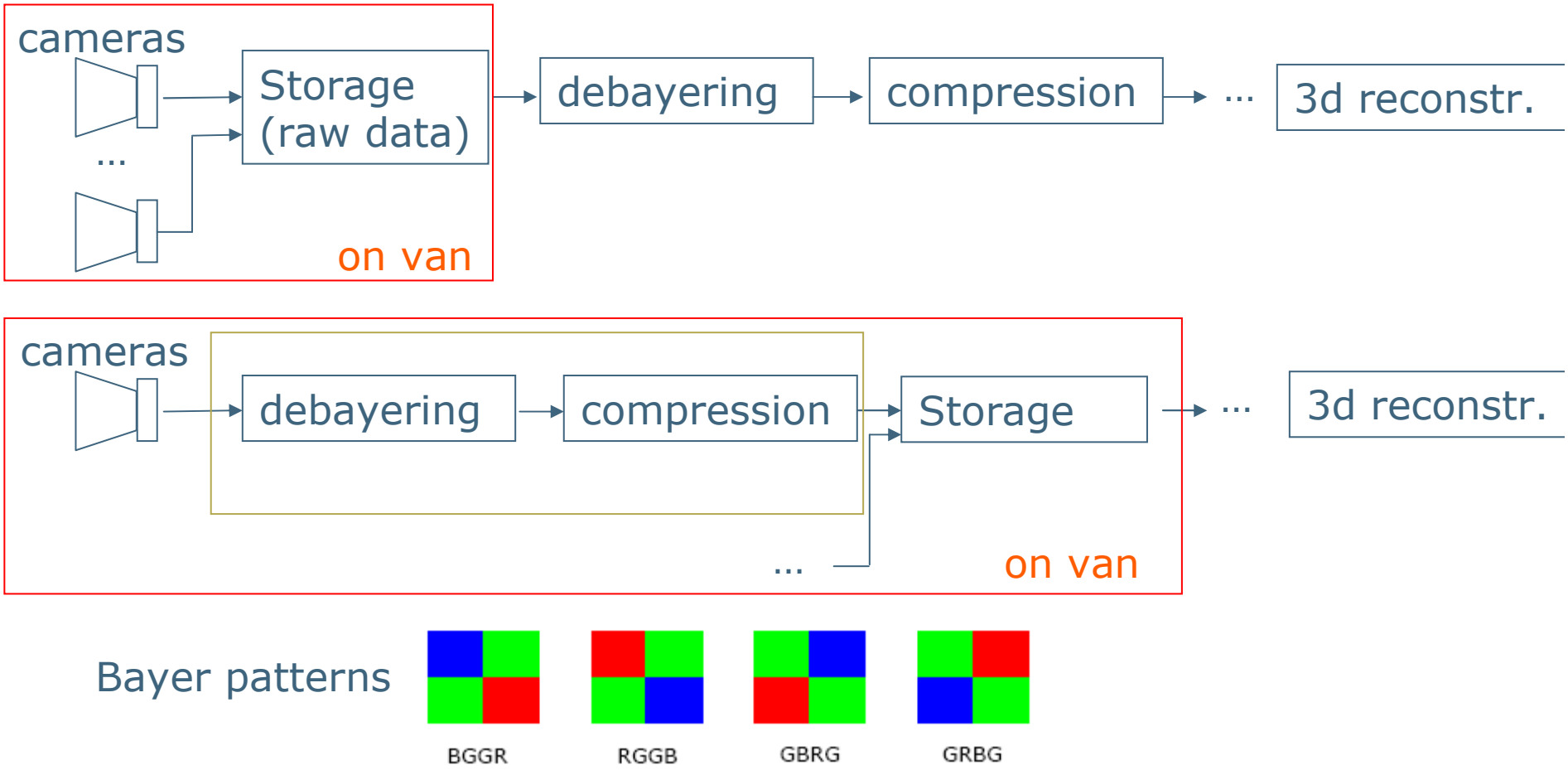
Task 1.1: On-board the recording van

Towards real-time HD video encoding: initial optimizations



Towards real-time HD video encoding

- Simplified acquisition pipeline: objective 12 fps

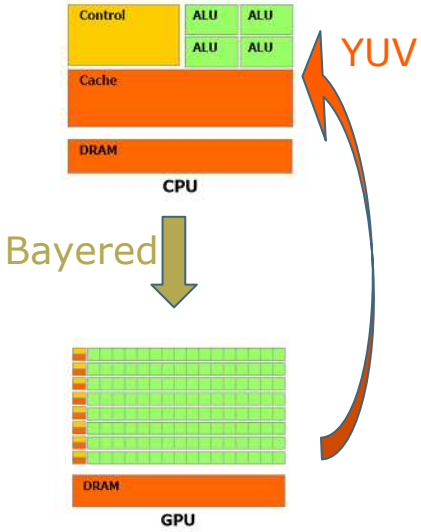


© imec restricted 2008

Towards real-time HD video encoding

- Measurements without optimizations
  - De-bayering : 40 fps
  - AVC encoding : 3 fps
- De-bayering is not the bottleneck but:

fps	De-bayering
CPU	40
CUDA	500
IPP	170



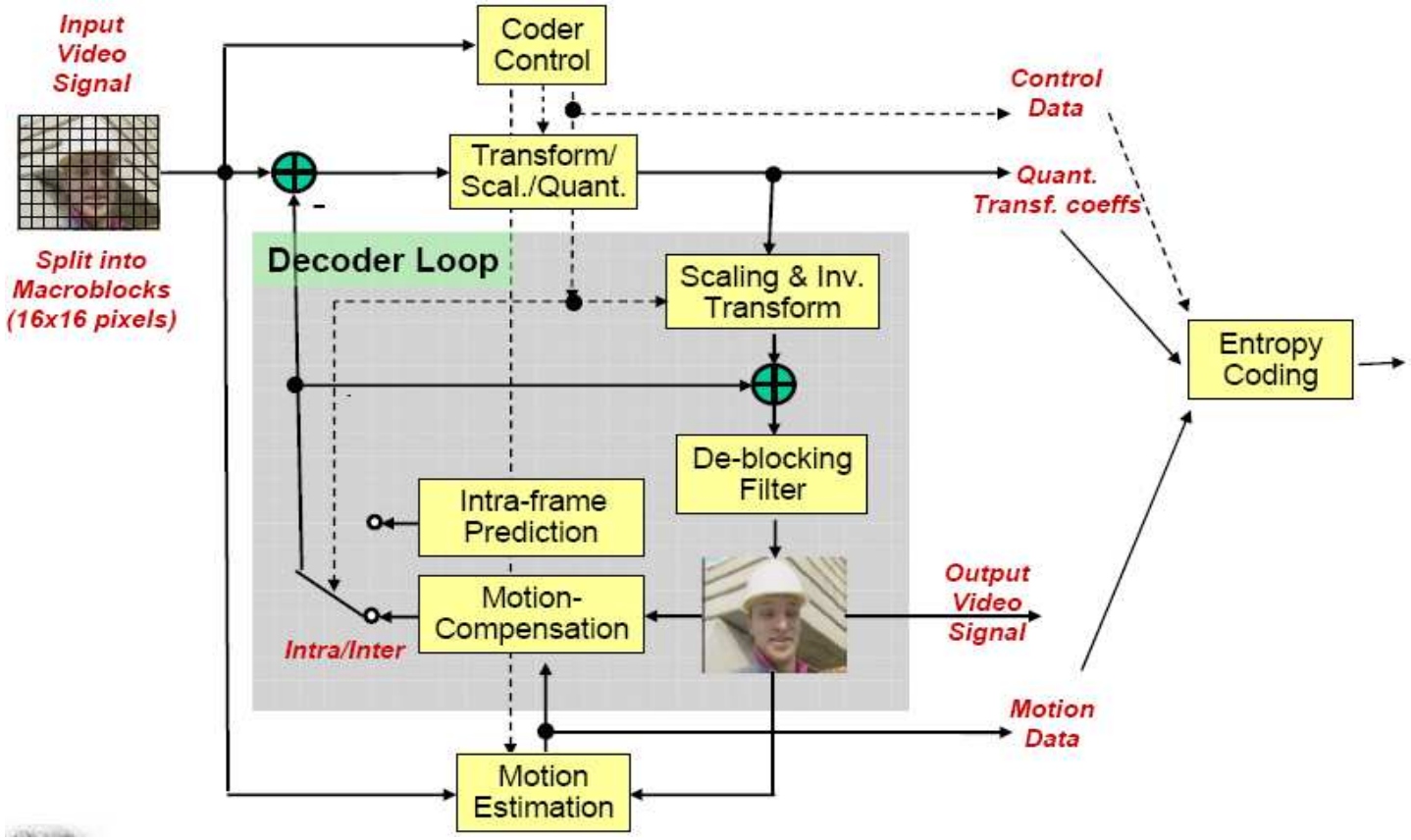
- CPU-GPU Bandwidth limitations!

CUDA	Bandwidth	De-bayering
YUV	250	195

- The best choice for a single functional unit is maybe not the most suitable for the end to end application

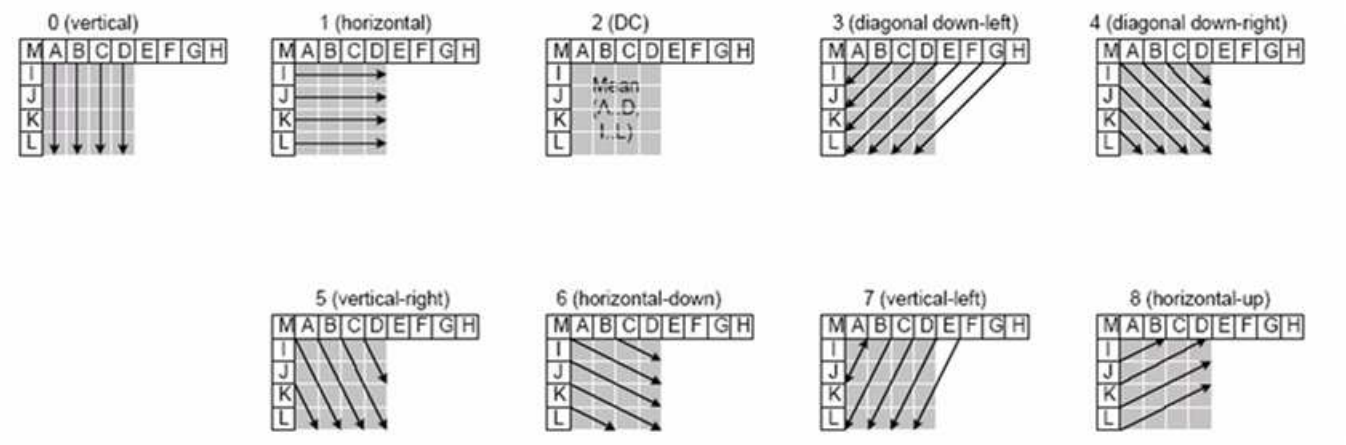
Towards real-time HD video encoding

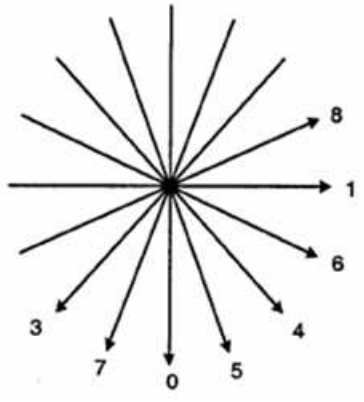
- H.264 AVC Intra



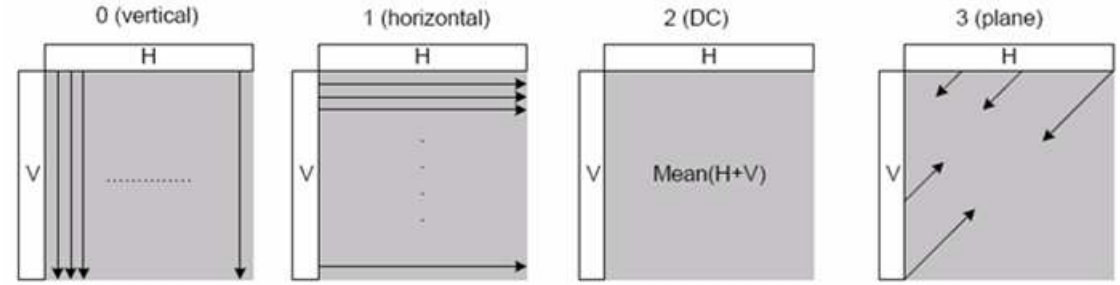
Towards real-time HD video encoding

- Intra prediction
  - 4x4





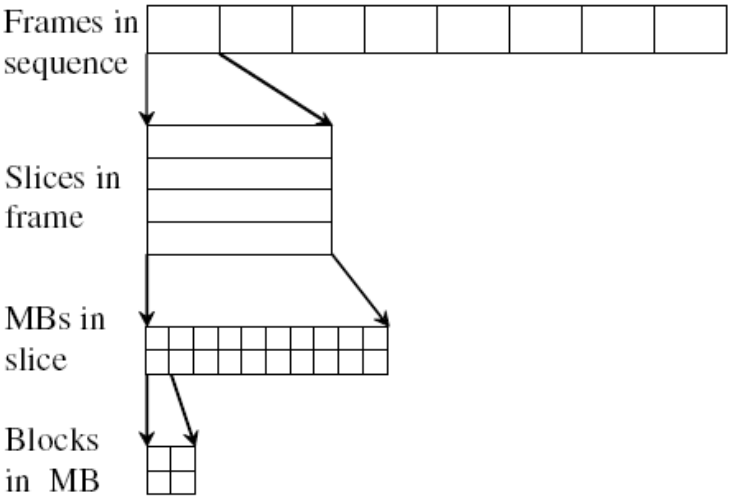
  - 16x16

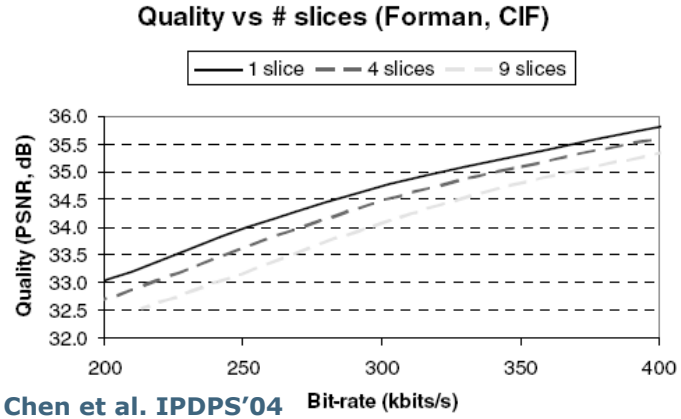


© imec restricted 2008

Towards real-time HD video encoding

- Parallelization Opportunities in H.264
  - Data hierarchical organization and data-domain decomposition
  - Intra coding : frames and slices are independently coded
  - Scalability
    - CPU: in frames but not in slices (rate distortion performance drops with the number of slices)
    - GPU: in blocks but dependencies
  - Load-balance
    - CPU and GPU: Intra "full" search enables better load-balance than sequentially optimized fast intra search





Chen et al. IPDPS'04

Towards real-time HD video encoding

- Using IPP blocks

module	Module Speed-up
SAD	3.5x
Integer Transform and Quantization	1.3x
Intra prediction (one mode)	1.4x
De-blocking filter	1.3x

- Using frame parallelism, preliminary results
  - With 4 CPUs:
    - speed-up 3.2x, 7fps
  - With 2 CPUs:
    - speed-up 1.8x, 4fps

© imec restricted 2008

Towards real-time HD video encoding

- Conclusion and Next steps
  - Combining frame and slice parallelism
  - Mapping to GPU with CUDA
  - Combining OpenMP and CUDA
  - Prototypes
    - M12:
      - D1.1.2: Implementation of the **flexible** compression framework using OpenMAX DL
    - M18:
      - D1.1.3: Implementation of an **optimal** compression engine that fits in the GeoAutomation tool-chain.
  - Task 2: Processing on the servers
    - Planning to improve feature detection with CUDA (SURF or SIFT)
    - Speeding-up plane-sweeping with CUDA

© imec restricted 2008


aspire invent achieve
imec