technische universität dortmund

computer science 12

Design Space Exploration Towards a Realtime and Energy-Aware GPGPU-based Analysis of Biosensor Data

Constantin Timm[1], Frank Weichert[2], Peter Marwedel[1] and Heinrich Müller[2]

[1] Department of Computer Science XII, TU Dortmund, Germany
[2] Department of Computer Science VII, TU Dortmund, Germany

Fakultät für Informatik

Contents

- Motivation
- Sensor and GPGPU-based Image Processing and Analysis Pipeline
- Design Space Exploration
- Results
- Conclusion

Motivation

- High-end graphics cards available in smaller and smaller systems → Concepts of HPC even in mobile systems
- GPGPU is standard for accelerating applications
- Energy efficiency is still often not a design objective

Motivation

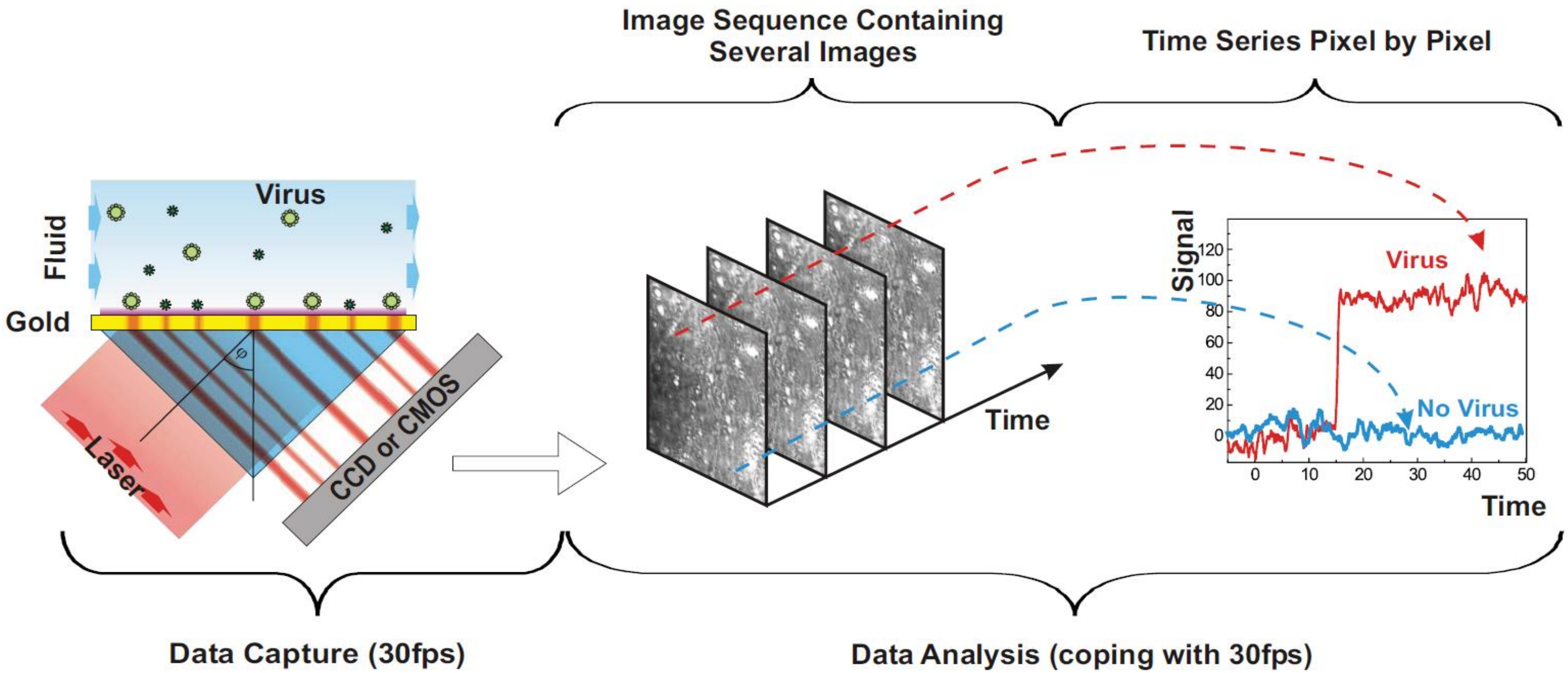
- At embedded system design, application set is fixed
  - Hardware/Software-Codesign optimizes applications to the (corresponding) platform
- In the past optimization was used in the HPC domain, only for achieving higher acceleration
  - But if it is not needed, can we save energy

?

Contents

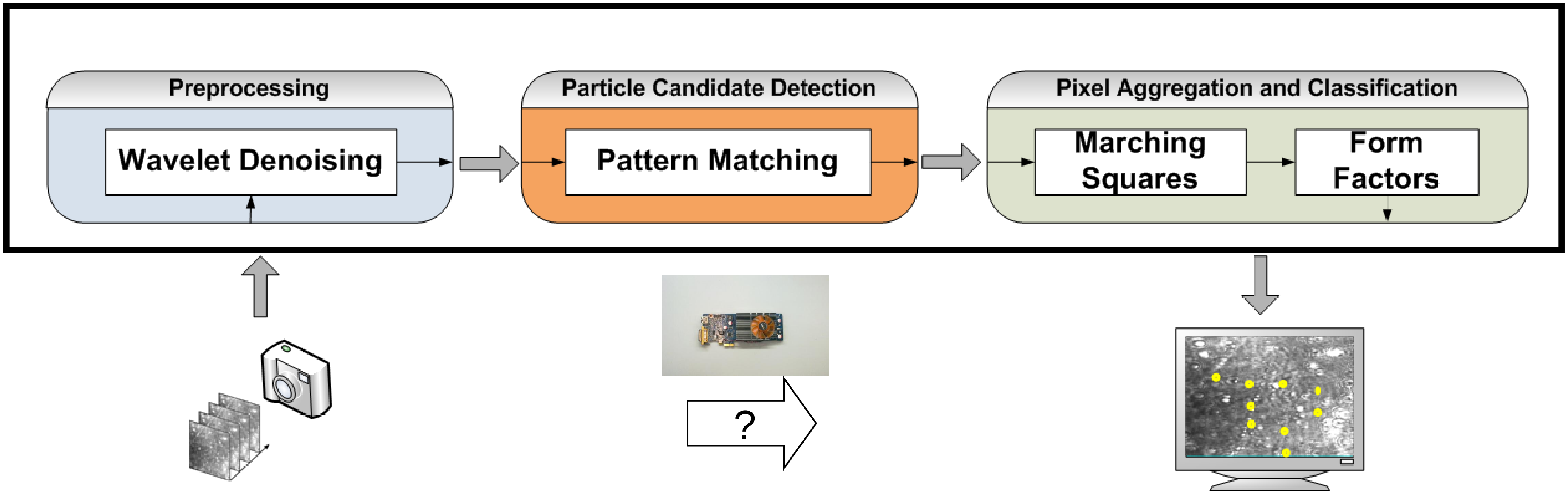
- Motivation
- Sensor and GPGPU-based Image Processing and Analysis Pipeline
- Design Space Exploration
- Results
- Conclusion

Microscopy/Image-Processing based Biosensor



- New sensor for detecting single viruses via microscopy
- Viruses have characteristic signal over time
- Sensor for in-situ disease spreading containment at airports etc
- Needs efficient image processing

Realtime Image Processing



- Physical effects ➔ certain camera speed ➔ speed of the image processing and analysis
- Image processing and analysis is done with GPGPU to meet realtime constraints
- Platform decision enables energy efficient solution

Contents

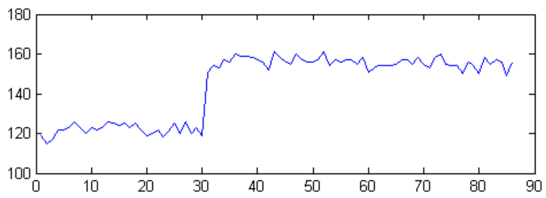
- Motivation
- Sensor and GPGPU-based Image Processing and Analysis Pipeline
- Design Space Exploration
- Results
- Conclusion

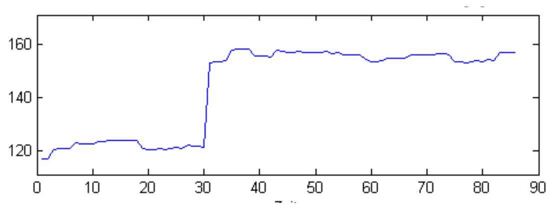
Design Space Exploration Overview

- DSE should reveal
  - an optimal system load configuration
  - an optimal system selection
- Two design parameters
  - Changing platforms / number of cores
  - Grouping of threads
- Two objectives
  - Processing speed / realtime requirements
  - Energy efficiency

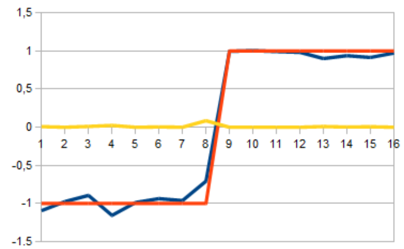
Scalability of Algorithms (1)

- Algorithms with high parallelization grade
  - Wavelet-based Denoising





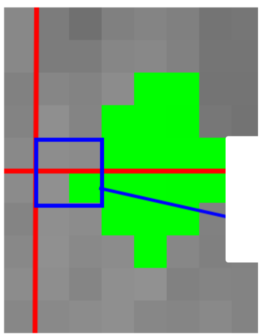
  - Template Matching (Virus Detection)

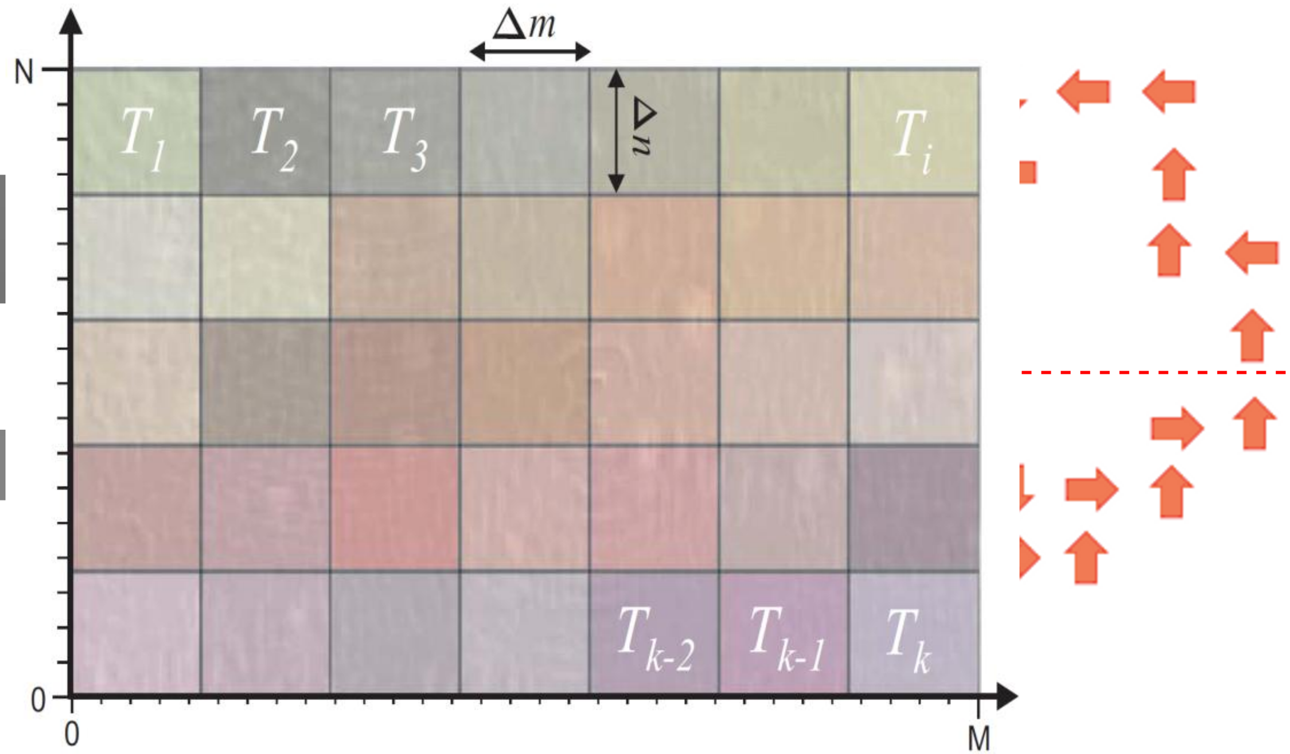


- Parallel processing of M*x*N time series for images with M*x*N pixels

Scalability of Algorithms (2)

- Algorithms with low parallelization grade: Marching-Squares



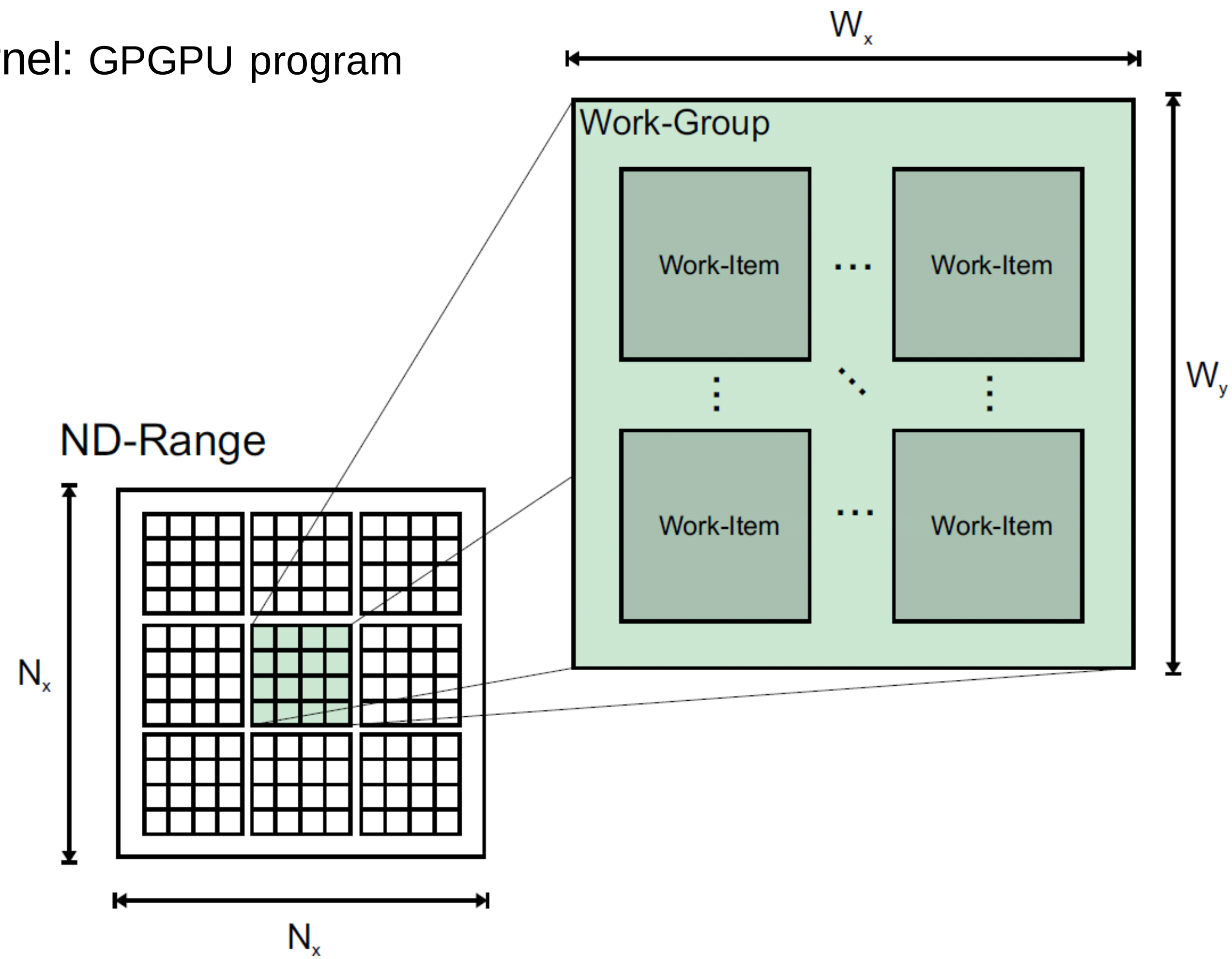


OpenCL: Nvidia Chip Design

- A (Nvidia) graphics chip comprises several streaming multi-processors (SM)
- Functional units of a SM:
  - Streaming processors (SP)
  - Shared Memory (16/32/64 K)
  - Registers (8192/16384/32768)

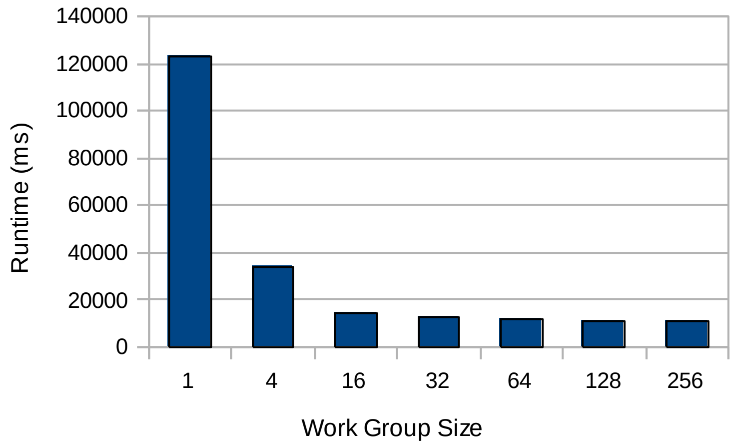
OpenCL: Programming Framework Elements

- Kernel: GPGPU program



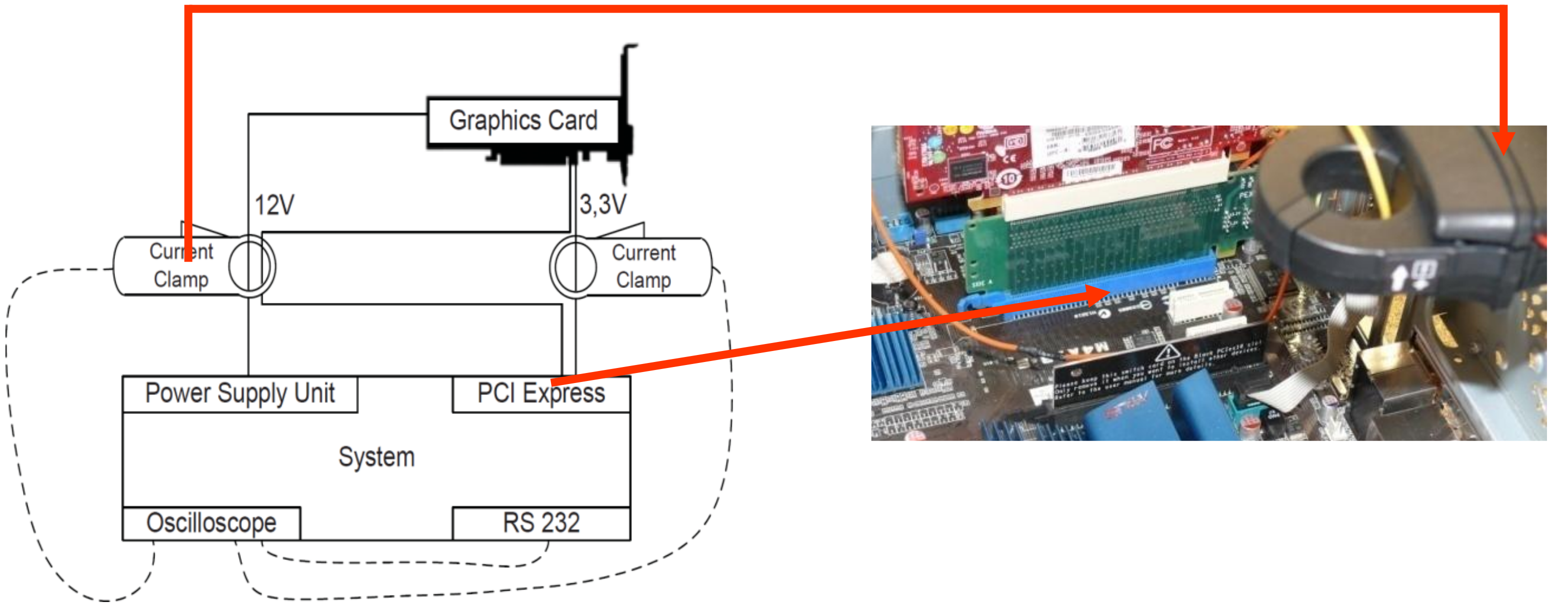
OpenCL: Work Group Size Considerations

- Large variety in runtimes
- Must be defined for each kernel
- Has impact on local memory use, scheduling etc
- Optimal size only optimal for one type of graphics card
- Optimal grouping is done by DSE, hard to predict



➔ Work Size Group is one parameter of the DSE

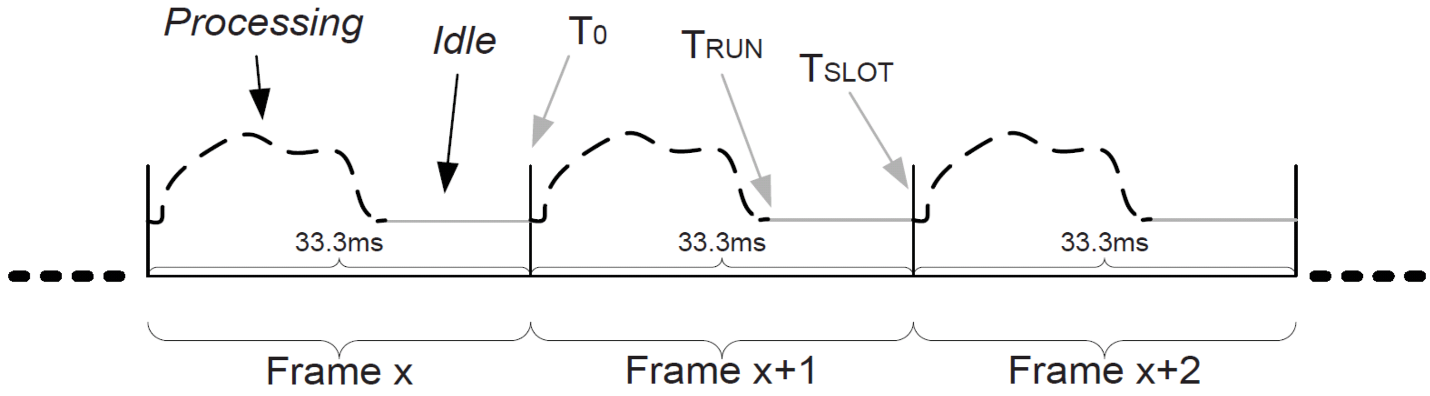
Objectives: Energy and Performance Testbed



- Standard PC
- Devices under Test: PCI Express graphics card
- Measurements: current clamp at PCI Express power supply lines
- Automatic analysis of GPGPU applications via trigger markers at source code level

Design Space Exploration Goal

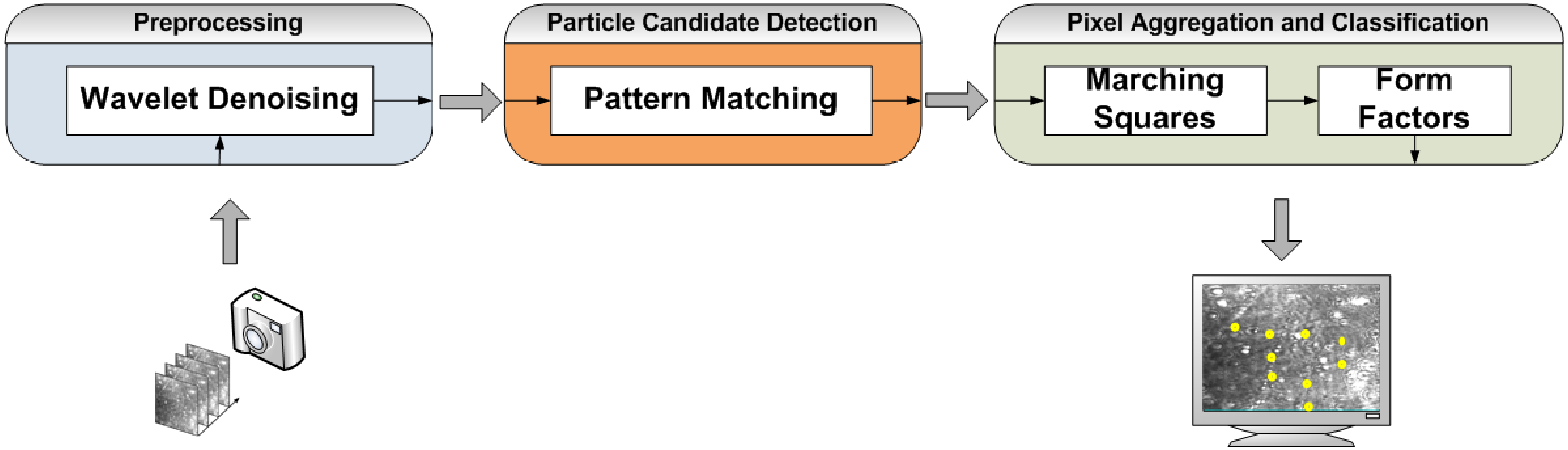
- Camera acquires 30 fps
- Image processing and analysis pipeline has to meet realtime constraints
- Energy efficiency is here defined as lowest energy consumption for one frame processing interval (~33 milliseconds)



Contents

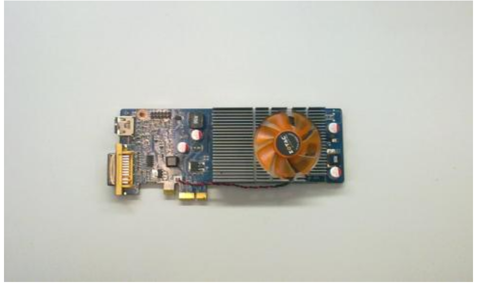
- Motivation
- Sensor and GPGPU-based Image Processing and Analysis Pipeline
- Design Space Exploration
- Results
- Conclusion

GPGPU Application Features



- 19 OpenCL kernels to optimize
- 12 kernels for frame processing
- Image upload before / image download after processing
- 7 kernels for initialization and shut down

Devices under Test



NextION

9600 GT

GTS 250

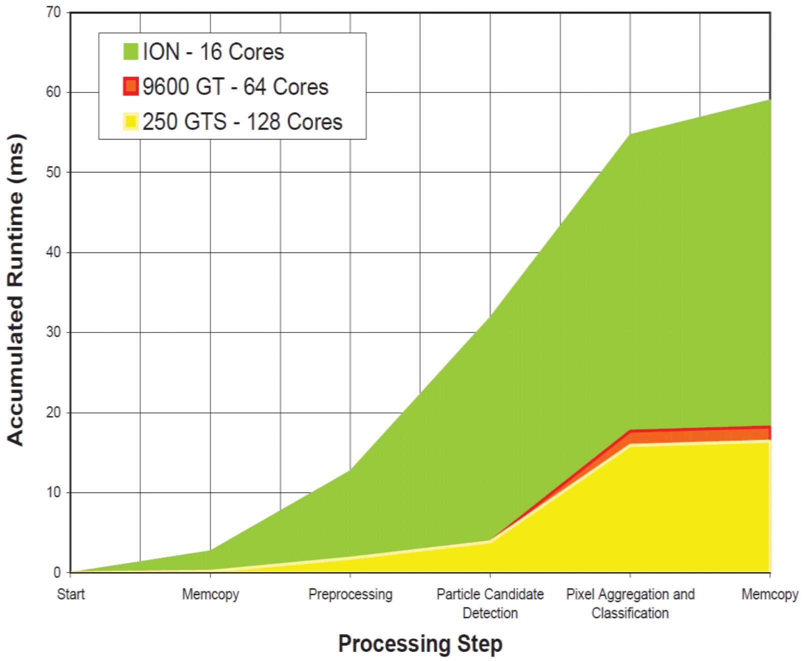
- Nvidia graphics cards with different size of cores

Graphics Card	Cores	Shader Clock (MHz)	Mem. Wid. (bit)	Idle Power (W)
NextION	16	1402	64	5.4 - 6.6
9600 GT	64	1625	256	33.48 - 43.2
GTS 250	128	1836	256	24.0 - 45.6

Results - Scalability

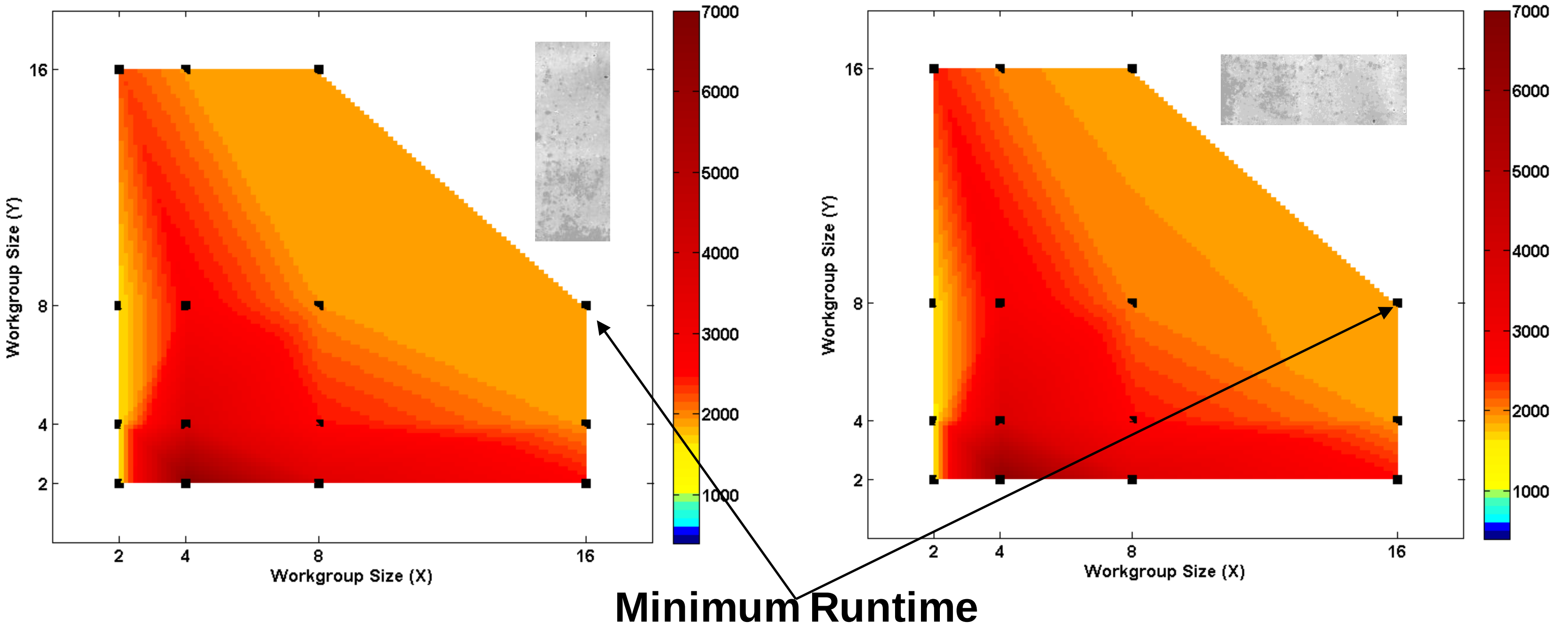
	ION → 9600 GT	ION → 250 GTS
Min	0.79	0.92
Max	5.92	9.31
Avg	2.88	3.43

- Min: Optimum <16
- Max: Superlinear scaling due to higher clock and memory speed

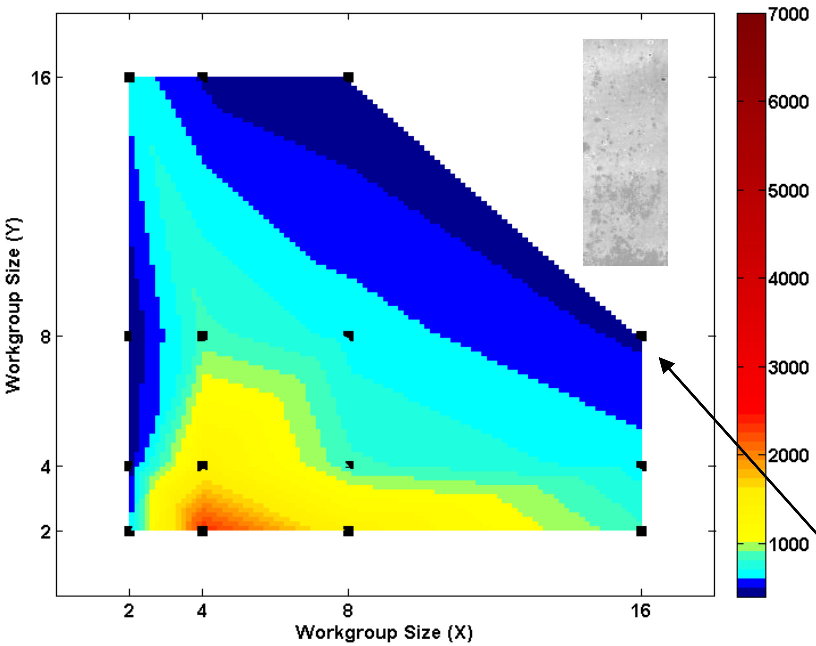


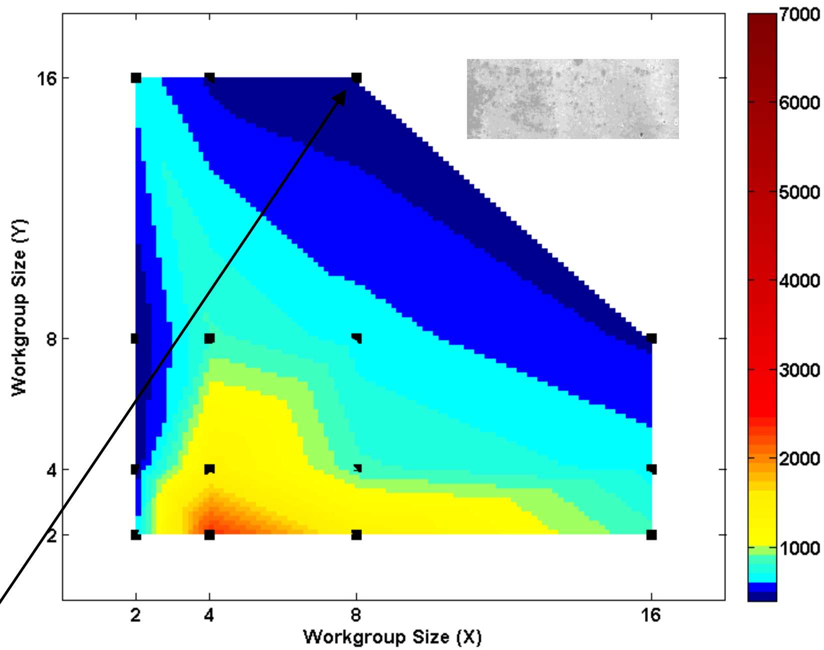
Results - Input Data Dependency

- 2 Validation Scenarios: Landscape and Portrait
- Kernel: Wavelet Denoising



Results - Input Data Dependency



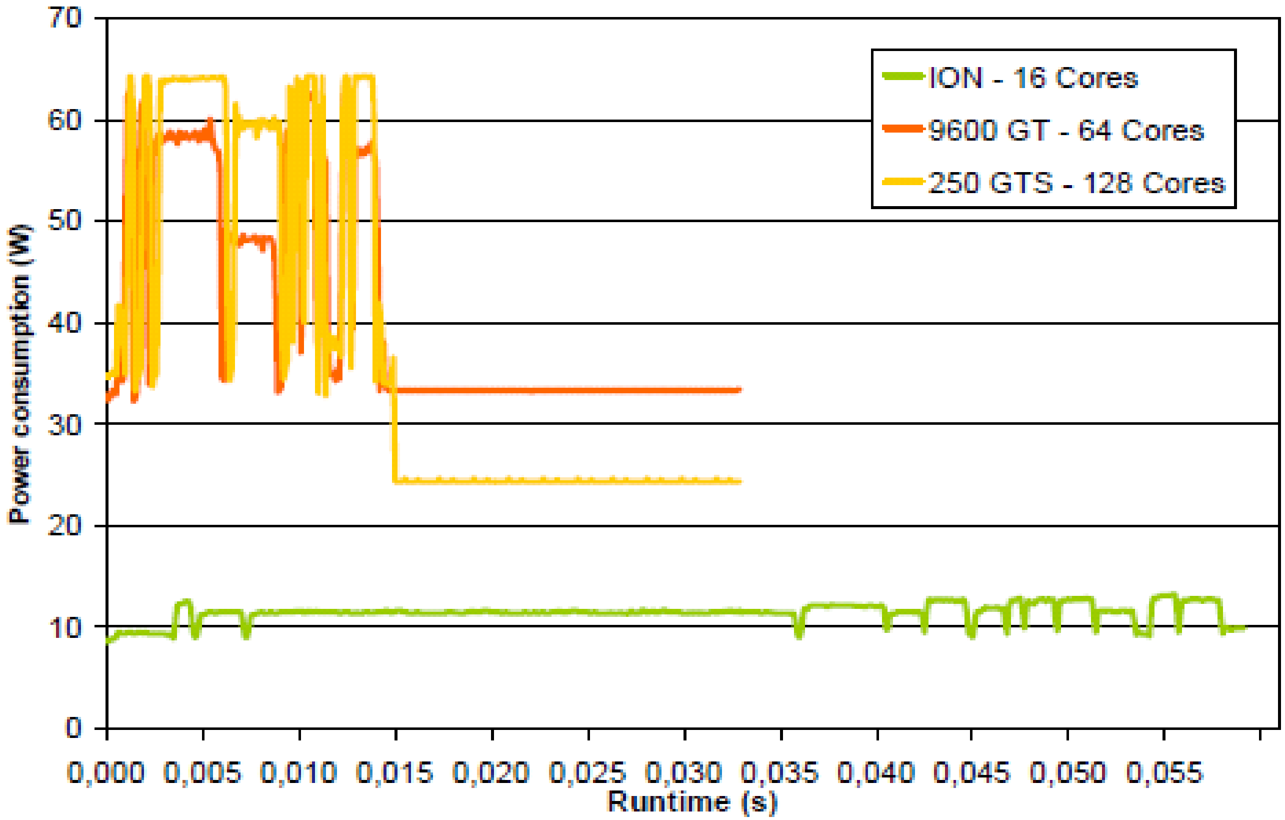


Minimum Runtime

- Kernels with many main memory accesses or a complicated control-flow
  - ➔ Different work group sizes have higher effect

Results - Power Consumption Over Time

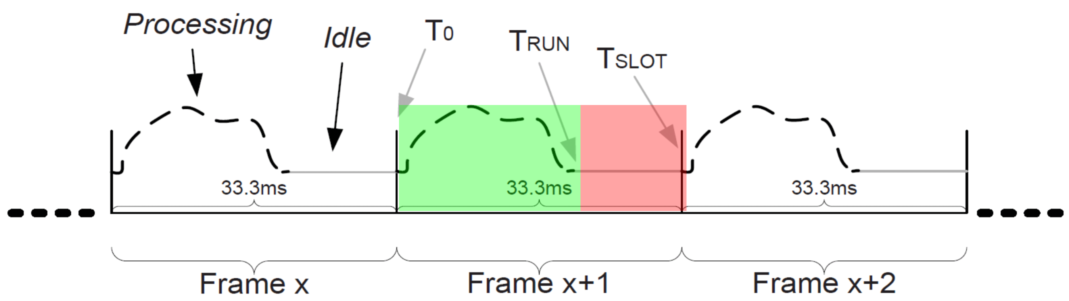
- Processing one frame



Results - Energy Consumption per Kernel (J)

Work Group Size		Wavelet Denoising			Pattern Matching			Form Factors		
X	Y	ION	9600 GT	250 GTS	ION	9600 GT	250 GTS	ION	9600 GT	250 GTS
16	2	**0.0381**	**0.0335**	0.0398	0.2973	0.2559	0.3148	0.0272	0.0347	**0.0330**
16	4	0.0392	0.0340	0.0400	0.2825	0.2103	0.2385	0.0263	0.0349	0.0349
16	8	0.0395	0.0340	0.0424	0.2900	0.2138	0.2455	0.0263	0.0368	0.0367
2	16	0.0415	0.0420	0.0542	0.2872	0.1976	0.2423	0.0306	0.0351	0.0348
2	2	0.0726	0.0723	0.0564	0.2877	0.2939	0.2501	0.0846	0.1228	0.0810
2	4	0.0536	0.0457	0.0409	**0.2457**	0.2000	0.2392	0.0530	0.0719	0.0508
2	8	0.0479	0.0363	0.0473	0.2822	0.1985	0.2531	0.0388	0.0473	0.0350
4	16	0.0407	0.0369	0.0443	0.2782	0.2002	**0.2335**	0.0267	0.0345	0.0341
4	2	0.0523	0.0451	**0.0389**	0.2837	0.2171	0.2662	0.0505	0.0714	0.0541
4	4	0.0458	0.0349	0.0411	0.2879	**0.1931**	0.2336	0.0362	0.0454	0.0360
4	8	0.0406	0.0355	0.0407	0.2885	0.2025	0.2529	0.0297	0.0353	0.0351
8	16	0.0399	0.0359	0.0434	0.2883	0.2120	0.2408	0.0263	0.0354	0.0359
8	2	0.0457	0.0344	0.0397	0.2990	0.2305	0.2878	0.0342	0.0460	0.0338
8	4	0.0394	0.0354	0.0429	0.2918	0.2055	0.2367	0.0277	0.0352	0.0344
8	8	0.0397	0.0361	0.0405	0.2826	0.2088	0.2454	**0.0262**	**0.0340**	0.0342

Results – Total Energy Consumption



Graphics Card	Runtime (ms)	$E^{g,p}_{T_0,T_{run}}$ (J)	$E^{g,p}_{T_{run},T_{slot}}$ (J)	$E^{g,p}$ (J)
ION	59.98	0.7	-	-
9600GT	18.16	0.95	0.39	1.34
250GTS	17.22	1.01	0.31	1.32

- Most energy efficient graphics card platform: GTS 250

Contents

- Motivation
- Sensor and GPGPU-based Image Processing and Analysis Pipeline
- Design Space Exploration
- Results
- Conclusion

Conclusions and Future Works

- Energy efficiency demands that an appropriate graphics card has to be chosen
- Energy saving techniques are mandatory to provide an energy-efficient system

- Dynamic voltage and frequency scaling could possibly be applied
- For more work group size configuration a DSE approach with e.g. an genetic algorithms should be used

Questions?