Real-Time Medical Imaging Using GPUs with a Non-Real-Time Operating System
X-ray Modalities @ Siemens Healthcare
Radiography

1 Images by Ysio Max
X-ray Modalities @ Siemens Healthcare
Mammography

Images by Mammomat Inspiration

1Images by Mammomat Inspiration
X-ray Modalities @ Siemens Healthcare
Surgery

Images by Cios Alpha

1Images by Cios Alpha
X-ray Modalities @ Siemens Healthcare
Surgery¹ (2)
X-ray Modalities @ Siemens Healthcare
Surgery¹ (3)
Motivation for Harmonized Image Chain (harmonIC\textsuperscript{1})

Former imaging solutions were designed for

• Single modality
• Dedicated image processing hardware (FPGAs and DSPs)
  • **Software solutions were not suitable for real-time image processing**

and **not** for

• Modularity and expandability
• Generality

[Establish new software solution for medical imaging.]

to support these novel requirements.

\textbf{harmonIC}\textsuperscript{1}

\textsuperscript{1}Working title, i.e. no Siemens brand
harmonIC – Profile

What is the “harmonized Image Chain”

- Software framework based on MS Windows
- Processes X-ray images from
  - acquisition via
  - image processing up to
  - presentation
- One communal software for all platforms
- Interface provides easy and abstract access to detector and image processing functionality
- Modular and object oriented approach

> 12 System Types

> 150 Algorithms

> 400,000 Lines of Code

> 3 Modalities

3 Image Systems

> 25 Contributors

17 Detectors and Cameras

Images by Cios Alpha and Ysio Max
harmonIC - Overview

Core
- Contains all communal functions like
- Acquisition and post-processing workflows
- Resource management
harmonIC - Overview

Core
• Contains all communal functions like
  • Acquisition and post-processing workflows
  • Resource management

Image Source Control
• Manages detectors and cameras in
  • Frame grabbing
  • Controlling
  • Triggering
harmonIC - Overview

Core
- Contains all communal functions like
  - Acquisition and post-processing workflows
  - Resource management

Image Source Control
- Manages detectors and cameras in
  - Frame grabbing
  - Controlling
  - Triggering

Image Processing Control
- Handles all image processing pipelines
harmonIC - Overview

Core
• Contains all communal functions like
  • Acquisition and post-processing workflows
  • Resource management

Image Source Control
• Manages detectors and cameras in
  • Frame grabbing
  • Controlling
  • Triggering

Image Processing Control
• Handles all image processing pipelines which
  • Access a CUDA based algorithm pool
Modular Pipeline Concept

- **Algorithm pool realized in software (CUDA)**
  → High performance realization of image processing
  → Efficient debugging and bugfixing of image processing

- **Platform specifics encapsulated in IP pipelines**
  → Pipeline changes does not interfere with other platforms
  → Used for acquisition and postprocessing (!!!)

- **Modular IP pipelines**
  → Easy integration of new algorithms
Modular Pipeline Concept

• **Algorithm pool realized in software (CUDA)**
  → High performance realization of image processing
  → Efficient debugging and bugfixing of image processing

• **Platform specifics encapsulated in IP pipelines**
  → Pipeline changes does not interfere with other platforms
  → Used for acquisition and postprocessing (!!!)

• **Modular IP pipelines**
  → Easy integration of new algorithms
Modular Pipeline Concept

• Algorithm pool realized in software (CUDA)
  → High performance realization of image processing
  → Efficient debugging and bugfixing of image processing

• Platform specifics encapsulated in IP pipelines
  → Pipeline changes does not interfere with other platforms
  → Used for acquisition and postprocessing (!!!)

• Modular IP pipelines
  → Easy integration of new algorithms
Modular Pipeline Concept

• Algorithm pool realized in software (CUDA)
  → High performance realization of image processing
  → Efficient debugging and bugfixing of image processing

• Platform specifics encapsulated in IP pipelines
  → Pipeline changes does not interfere with other platforms
  → Used for acquisition and postprocessing (!!!)

• Modular IP pipelines
  → Easy integration of new algorithms
Modular Pipeline Concept

- **Algorithm pool realized in software (CUDA)**
  - High performance realization of image processing
  - Efficient debugging and bugfixing of image processing

- **Platform specifics encapsulated in IP pipelines**
  - Pipeline changes does not interfere with other platforms
  - Used for acquisition and postprocessing (!!!)

- **Modular IP pipelines**
  - Easy integration of new algorithms
Modular Pipeline Concept

- **Algorithm pool realized in software (CUDA)**
  → High performance realization of image processing
  → Efficient debugging and bugfixing of image processing

- **Platform specifics encapsulated in IP pipelines**
  → Pipeline changes does not interfere with other platforms
  → Used for acquisition and postprocessing (!!!)

- **Modular IP pipelines**
  → Easy integration of new algorithms
Real-Time Challenges I

Time Lag

Presenting the acquired X-ray (video) as fast as possible is crucial.

Main time intervals:
- X-ray detector → framegrabber
- Framegrabber → Host memory (no GPUDirect)
- Host memory → Image Processing on GPU
- Image Processing → Display
Real-Time Challenges I

Time Lag

Presenting the acquired X-ray (video) as fast as possible is crucial.

Main time intervals:
- X-ray detector → framegrabber
- Framegrabber → Host memory (no GPUDirect)
- Host memory → Image Processing on GPU
- Image Processing → Display

Measure:
- Page-locked memory
- Render the processed image on the same GPU without additional copy
Real-Time Challenges II

Constant Framerate

Rendering the X-ray video jitter-free is essential.

Problems:
• Different clock-rates of involved components (detector, monitor)
• Non-real-time “components”
Real-Time Challenges II

Constant Framerate

Rendering the X-ray video jitter-free is essential.

Problems:
• Different clock-rates of involved components (detector, monitor)
• Non-real-time “components”

Acquisition:
• Clock-pulse generator is the image source

Replay of X-ray video:
• Clock-pulse generator is the monitor
Real-Time Challenges III

Stability

Presenting the acquired X-ray (video) as stable as possible is fundamental.

Problems:

- OS-related unsteadiness
- Connected Image System software
- Hardware-interrupts (not solved yet 😞)
Real-Time Challenges III

Stability

Presenting the acquired X-ray (video) as stable as possible is fundamental.

Problems:
- OS-related unsteadiness
- Connected Image System software
- Hardware-interrupts (not solved yet 😞)

Two ring buffers:
- Acquisition-buffer for OS-related jitter
- Processed-image-buffer for Image System SW

Prevent radiation exposure without imaging:
- Two-stage escalation strategy
Upcoming

- GPU sharing across components
  - 2D image processing and visualization
  - 3D reconstruction
  - 3D volume visualization

- Parallelization of system workflows
  - Acquisition and replay at once
  - Critical vs. non-critical tasks

→ NVIDIA, help! 😊
Questions?

Stefan Schneider
Siemens Healthcare GmbH
Diagnostic Imaging
X-Ray Products
Research & Development
HC DI XP R&D IC IP
Allee am Roethelheimpark 2
91052 Erlangen
Germany
Phone: +49 (9131) 84-3449
E-mail: schneider.stefan@siemens.com