Go Faster - Preprocessing Using FPGA, CPU, GPU

Dipl.-Ing. (FH) Bjoern Rudde
Image Acquisition Development
STEMMER IMAGING
WHO ARE STEMMER IMAGING?

STEMMER IMAGING is:

- Europe's leading independent provider of
  - Core vision technology
  - Solutions
  - Services

Our mission:

- To provide the users and developers of imaging technology with competitive advantage by adding value in the supply of quality components, expertise and support.
To provide the users and developers of imaging technology with competitive advantage by adding value in the supply of quality components, expertise and support.
THE MARKETS WE SERVE

- Factory Automation
- Automotive
- Electronics, Semiconductor & Solar
- Print & Packaging
- Food & Beverage
- Pharmaceutical
- Test & Measurement
- Medical Imaging
- Traffic, Rail & Transport
- Scientific Research
- Defense, Security & Aerospace
- Sports, Entertainment & Broadcast
INCREASING DATA RATES

GO FASTER
Preprocessing using FPGA, CPU, GPU

In all markets there are applications that demand high data rates.

- Image resolution
- Frame rate
- Pixel depth
- Number of operations per pixel

Each parameter increases computing efforts.
DECISION FOR AN ARCHITECTURE

Vision applications have widely varying requirements and no single approach is ideal for each and every application. Therefore different architectures have their place.

- FPGA
- CPU
- GPU
CPU

Component found in each computer system and most machine vision libraries are based on this architecture.

- Sequence of single instruction on single data (SISD)
- Extended by vector processing instructions (SIMD)
  - MMX
  - ISSE
  - 3DNow!
  - AltiVec
  - SSE2, SSE3, SSSE3, SSE4, SSE5
- Multiple cores
- Operations / s defined by tick frequency
- One to several ticks needed per instruction
In the field of machine vision, a graphics processing unit (GPU) and its highly parallel structure can be used for uniform calculations on large image blocks.

- Single instruction on multiple data (SIMD)
- Memory intensive operations
- Massive floating-point computational power
- Fast development due to gaming market
- In most cases 1 tick / operation / proc. unit
- General purpose GPU (GPGPU)
FPGA

Field programmable gate arrays can be found in many imaging devices, but are not an inherent part of a computer system. This architecture is a reconfigurable integrated circuit.

- Multiple instructions on single data (MISD)
- High data throughput
- Massive parallelism
- Flexibility in spatial and temporal parallelism and tick frequency
- Direct hardware access (Trigger I/O, CC)
DEFINING PRE-PROCESSING STEPS

The individual processing steps are defined/identified during a feasibility study.

► First approach is based on software tools (CPU)
► Effort estimation of each step
► If computational power of a CPU-System is sufficient
  yes : select CPU
  no  : profiling of processing steps
► Evaluate FPGA or GPU solution
OPERATOR TYPES

Nearly all standard pre-processing functions fit into 5 general groups:

- Pixel operations: (in)homogeneous
- Histogram or LUT based functions
- Neighbour operations
- Random access operations
- Geometrical transformations
**ARCHITECTURE AND OPERATION**

**PERFORMANCE**

Architectures versus their general suitability for the different kinds of operators:

<table>
<thead>
<tr>
<th>Operation</th>
<th>CPU</th>
<th>GPU</th>
<th>FPGA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pixel</td>
<td>+</td>
<td>++</td>
<td>++</td>
</tr>
<tr>
<td>Histogram / LUT</td>
<td>++</td>
<td>-</td>
<td>++</td>
</tr>
<tr>
<td>Neighbour / Kernel</td>
<td>-</td>
<td>++</td>
<td>++</td>
</tr>
<tr>
<td>Random Access</td>
<td>++</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Geometrical Transform</td>
<td>+</td>
<td>++</td>
<td>-</td>
</tr>
<tr>
<td>Architecture</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SISD</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SIMD</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MISD</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
ARCHITECTURES IN ACQUISITION CHANNEL

Each architecture for pre-processing may be used in different positions of the acquisition channel:

- Source / Camera FPGA
- Output / Interface FPGA
- Input / Grabber FPGA
- Host system CPU / GPU
### PERFORMANCE CONSIDERATIONS

Different performance characteristics need to be considered with caution:

<table>
<thead>
<tr>
<th>Unit</th>
<th>CPU</th>
<th>GPU</th>
<th>FPGA</th>
</tr>
</thead>
<tbody>
<tr>
<td>GFLOPS</td>
<td>107</td>
<td>~5000</td>
<td>t.b.a.</td>
</tr>
<tr>
<td>GOPS</td>
<td>153</td>
<td>~5000</td>
<td>~125000</td>
</tr>
<tr>
<td>RAM GB/s</td>
<td>26</td>
<td>320</td>
<td>17</td>
</tr>
<tr>
<td>cache / pipeline</td>
<td>111 / -</td>
<td>- / -</td>
<td>- / 4000</td>
</tr>
<tr>
<td>RAM GB</td>
<td>&gt; 16</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>DMA GB/s</td>
<td>n.a.</td>
<td>4</td>
<td>3,6</td>
</tr>
</tbody>
</table>

GPU: AMD Radeon HD 6990 (end user product)
CPU: Intel Core i7 X 980 (4611MHz)
FPGA: SiliconSoftware microEnable 5, Xilinx Virtex 6 series ~250MHz
Evaluate FPGA or GPU solution

If today’s CPUs’ computational power is insufficient for the required processing steps.

- Architectures adaptable
- Pre-Processing steps convertible
- Flexibility and delay satisfactory
- Development tools available
- Time to market
- Combined processing option
## POSSIBLE PROCESSING FUNCTIONS

<table>
<thead>
<tr>
<th>Operation</th>
<th>GPU</th>
<th>FPGA</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Pixel</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Homogenous</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Binarization</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Color transform</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Inhomogenous</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Shading, Dynamic Thresholding</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td><strong>Histogram / LUT</strong></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Histogram</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Filters</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sobel, Median, FIR</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Segmentation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Erode, Dilate</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Transformation</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DCT, DFT, FFT, Wavelet</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Geometrical</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hough</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Compression</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>jpeg-encoding</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>H.264</td>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>

**Direct GPIO & Trigger control**: possible

**Integration into image source / acquisition interface**: possible
FLEXIBILITY AND DELAY SATISFACTORY

FPGA pre-processing capabilities can be found in frame grabbers and cameras. Different interfaces may require porting of the processing.

**FPGA**

- CameraLink, GigE, CoaXpress, CL HS
- Low power consumption
- Expandable / scalable
- Practically no delay
- Long term availability

**GPU**

- No acquisition device
- Adaptable by DMA
- Multi-GPU (SLI, Xfire, …)
- Minimal delay caused by DMA transfer
- Portable implementation possible
DEVELOPMENT TOOLS AVAILABLE

Converting processing steps to a different hardware is currently not possible by just compiling it for a different architecture.

- Imaging libraries supporting conversion
  - Common Vision Blox 2011, etc.
- GPU
  - CVB + DirectX, OpenCL
  - CUDA, StreamingSDK
- FPGA
  - Silicon Software / Visual Applets & eVA
  - Teledyne DALSA / Sapera APF
  - NI / LabView
  - MATLAB / Simulink
Most acquisition streams can utilize several architectures. Using these in combination for implementing image processing is an applicable and reasonable way.

- Camera (FPGA)
  - Acquisition device (FPGA)
    - Computer system (CPU & GPU)
  - architectures are currently merging
    - CPU + GPU
    - FPGA + CPU
    - Bright prospects
EXAMPLES CAN BE FOUND IN DIFFERENT FIELDS

All of the following examples are realized projects.

- FPGA for filtering & segmentation, CPU for classification in linescan
- FPGA for synchronizing several linescan cameras and merging frames
- GPU for filtering images
- 7 grabbers using FPGA processing in parallel
- FPGA for jpeg compression in high speed recording
- GPU for perspective correction & de-Bayering, CPU for compression
- See 2 of our demos including FPGA (pre-)processing, hall 4, booth C51
IN RELIABLE PARTNERSHIP WITH YOU

Intelligently integrated components and perfect service add value to your business.

- Optimised components
- Specific development
- Partner programs
- Advanced integration

CALL
- Pre-sales service
- On-site consultation service
- Feasibility studies
- Recommendation of independent system integrators

ORDER
- Quick and accurate order processing

DELIVERY
- Reliable delivery from Europe’s largest stock of imaging technology
- Highly attractive pricing/volume discounts
- 24-hour service for many products

IMPLEMENTATION
- Test run at customer site
- Implementation by system integrator/by customer
- Training at STEMMER IMAGING training centre/on customer site

DIALOGUE
- After-sales support by experienced engineers
- Timely services by expert staff
- Additional knowledge transfer via different media
OUR PRODUCTS

- Illumination
- Optics
- Cameras
- Cabling
- Acquisition
- Software
- Systems
- Accessories
Thank you for your attention!

STEMMER IMAGING GmbH
Gutenbergstraße 9 – 13
82178 Puchheim, Germany
Phone: +49 89 80902-294
Fax: +49 89 80902-116
info@stemmer-imaging.de
www.stemmer-imaging.de

Dipl.-Ing. (FH) Bjoern Rudde
Image Acquisition Development
b.rudde@stemmer-imaging.de