Seam Carving Through Time

Team BD: Maxwell Johnson, Riki Khorana, John Zhang

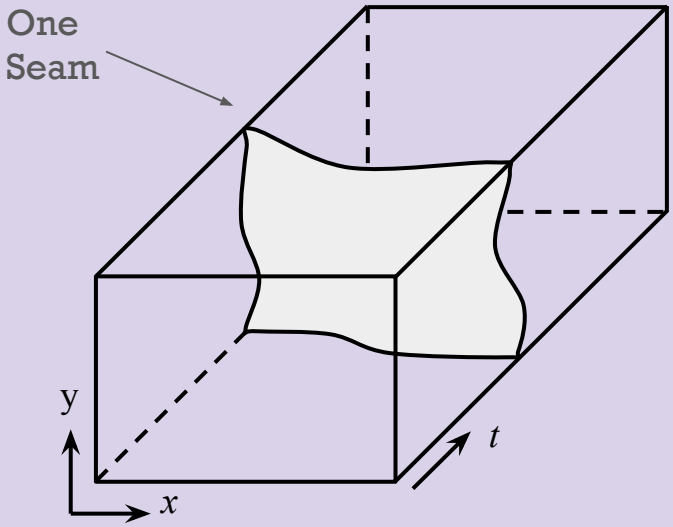
Previously on SCTT...

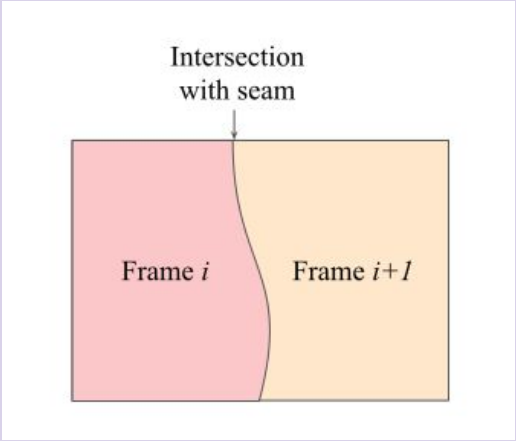
- Seam Carving
  - An algorithm for content-aware image resizing
    - Calculate Energy Map(pixel difference)
    - Find Lowest Energy Seam
    - Remove the Seams

Seam Carved

Our Project

- Seam Carving videos across time
  - Content-aware alteration of video playback speed
    - Find 3D energy map
    - Find 2D lowest energy seams
    - Remove the seams to shorten videos
    - Result: Output video frames comprise pixels from multiple frames
  - Speed up by delegating homogenous computation to FPGA



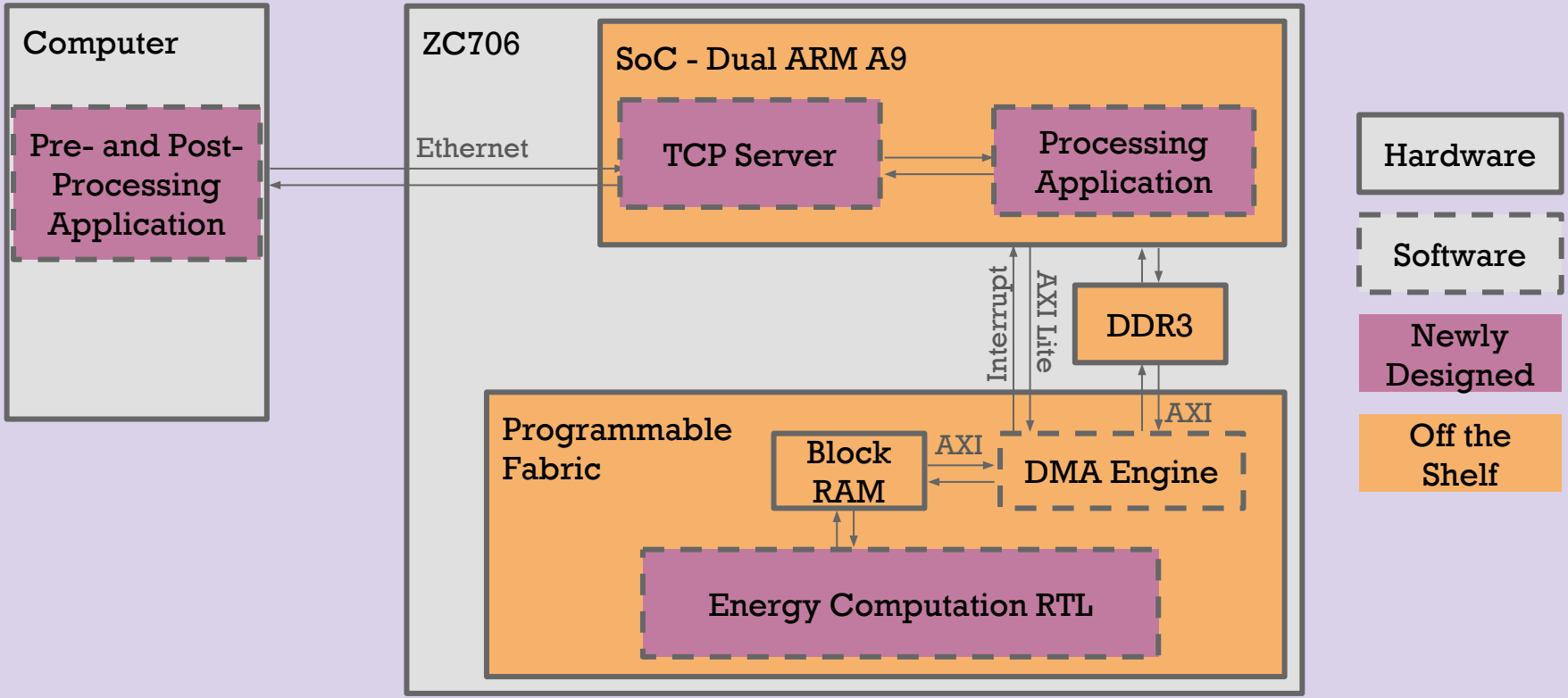


Our Latest Output Video

Input video:
Length: 3.7s

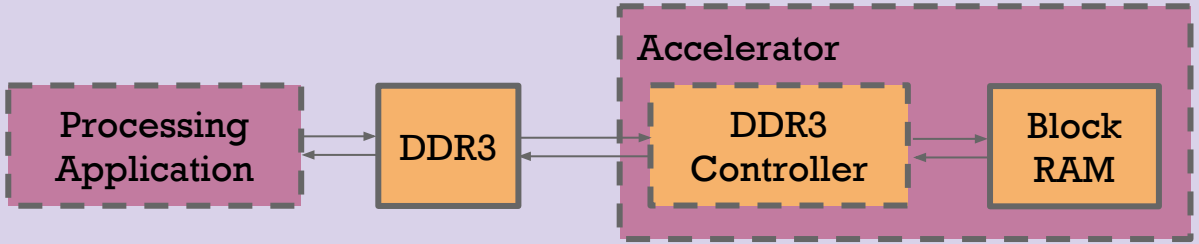
Output video:
Length: 2.4s

Final Implementation

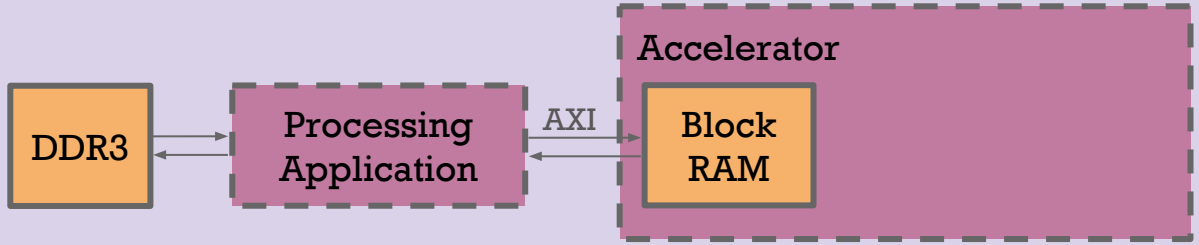


Major Design Changes – Memory Transfer

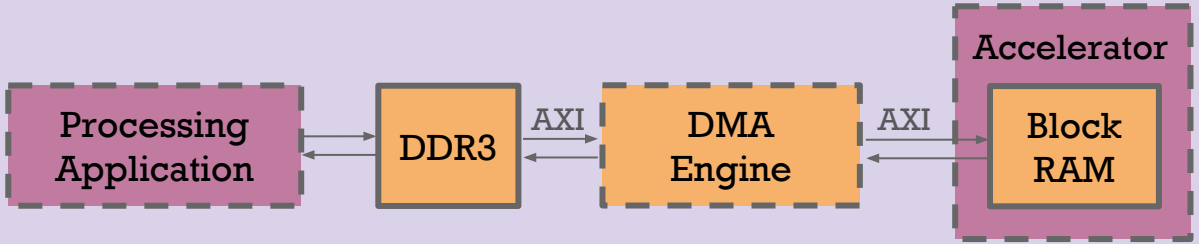
- Original Design



- Midpoint Demo



- Final Design [WIP]



Major Design Changes – Algorithm

1.) Graph-cut algorithm

+ Ensures optimal seam (min-cut)
+ Properly implements “forward energy”
- Graph size too big for FPGA

2.) Dynamic programming (sweep)

+ Takes less space
- Right-most column dominates
- Seams are discontinuous

3.) Dynamic programming (radix)

+ Seams are continuous
+ Properly prioritize areas with less action

Design Changes - Energy Function

Conventional Energy Mapping

“Forward Energy” Mapping

System Metrics

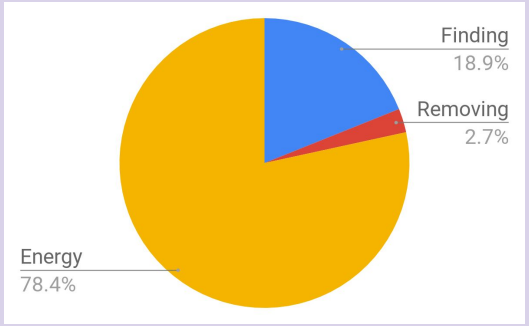
Process	Metric	Where we currently are
Increase playback speed by 1.5x	Frame count	Playback speed goal reached
Video is smooth	Subjective appraisal, energy function	Qualitative: Great Quantitative: TBD
Send byte stream of seam carved video to PC	Bits / second	20 Mbps
Process a video of duration T in time 3T	On-board timing from processing start to end	For 3.7s video: 184s [WIP]

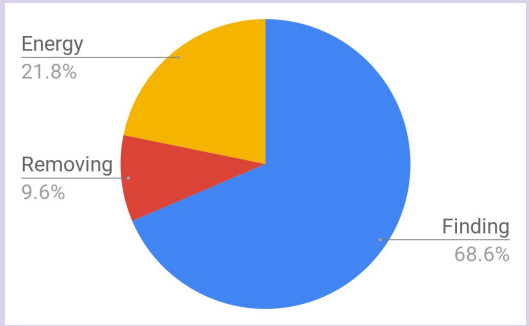
Timing

Reference time: 22 s (Macbook Pro, 2.9 GHz i5)

- Receive from computer: 10.5 s
- Total compute time: 184.6 s
  - Total time finding seams (SoC): 34.9 s
  - Total time removing seams (SoC): 4.9 s
  - Total time computing energy (PL): 144.6 s*
- Transmit to computer: 8.4 s

*Energy computing time using DMA: 11.1 s





Schedule

Week 1	PHASE 1: Design
Week 2-6	PHASE 2: Research + Implementation + Testing on individual modules
Week 7-10	PHASE 3: Integration
Week 11-12	PHASE 4: Testing + Application Survey, Optimization, Final Demo

Lessons Learned

- Know your tools
  - Vivado learning curve
  - Platform architecture
  - Integration first
- Understand the timeline
  - Don’t put the horse before the cart
  - Optimism is not your friend
- Trade-offs
  - Implementation time is part of the equation