Why Do GPUs Work So Well for Acceleration of CT?

SPIE Electronic Imaging '07

Keynote, Computational Imaging V

Klaus Mueller   Fang Xu

Computer Science

Center for Visual Computing

Stony Brook University

First: A Big Word of Thanks!

… to the millions of computer game enthusiasts worldwide

Who demand an utmost of performance and realism of their game engines

And who create a market force for high performance computing that beats any federal-funded effort (NSF, DOE, NASA, etc.)

High Performance Computing on the Desktop

Stony Brook Center for Visual Computing

PC graphics boards featuring GPUs:

- NVidia FX, ATI Radeon
- available at every computer store for less than $500
- set up your PC in less than an hour and play

NVIDIA GeForce FX

POWERED BY ATI RADEON 9600 PRO

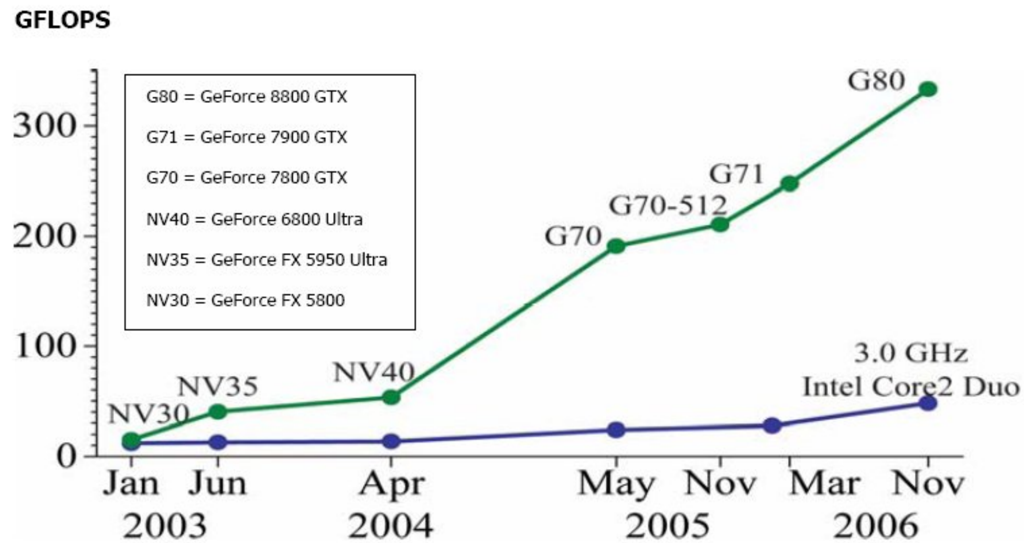
the latest board:

Nvidia GeForce 8800 GTX (G80)

Incredible Growth

Performance doubles every 6 months!

- triple of Moore's law

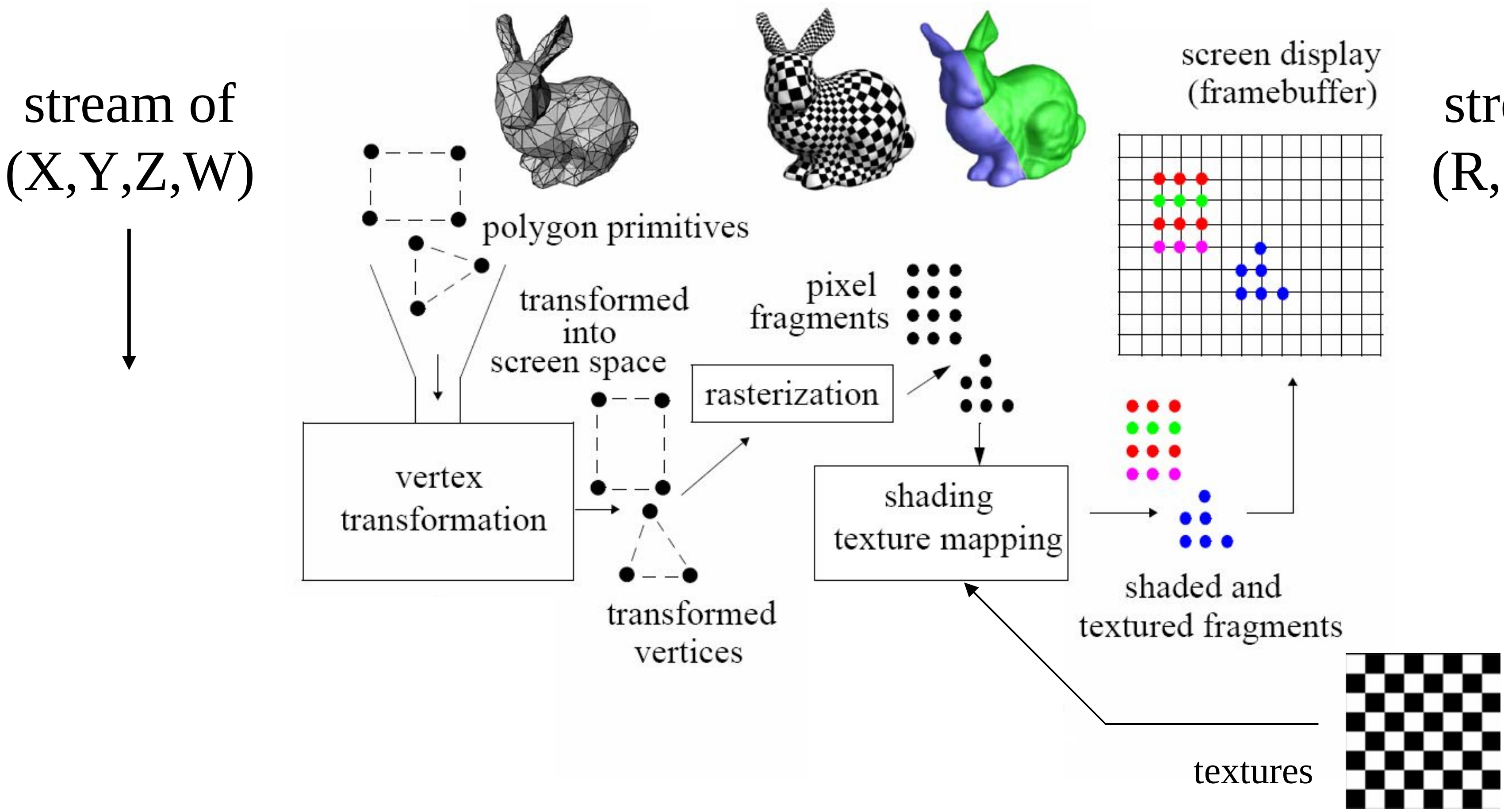


Performance gap GPU / CPU is growing

- currently 1-2 orders of magnitude is achievable
  (given appropriate programming and problem decomposition)

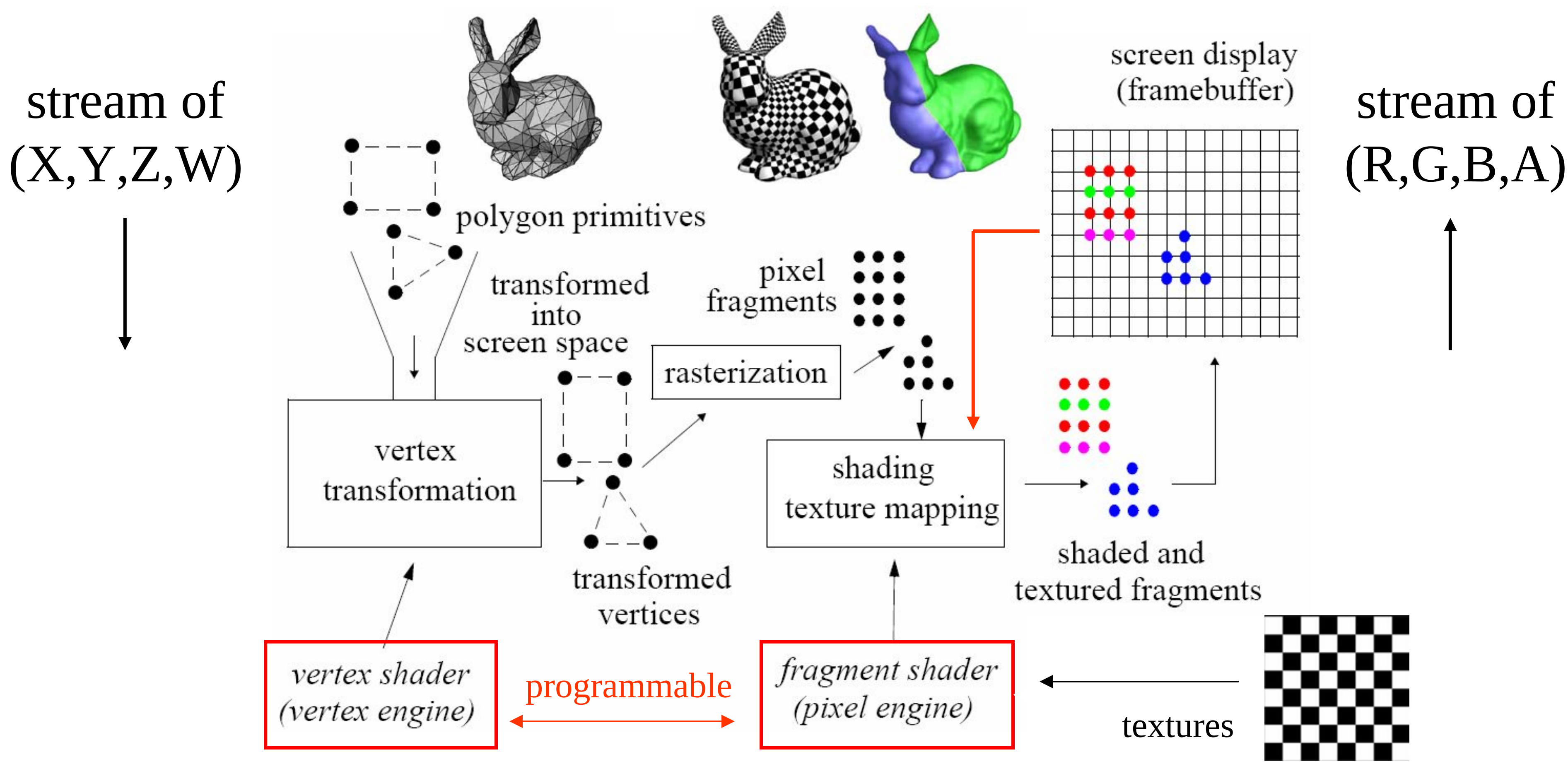
The Graphics Pipeline

Old-style, non-programmable:



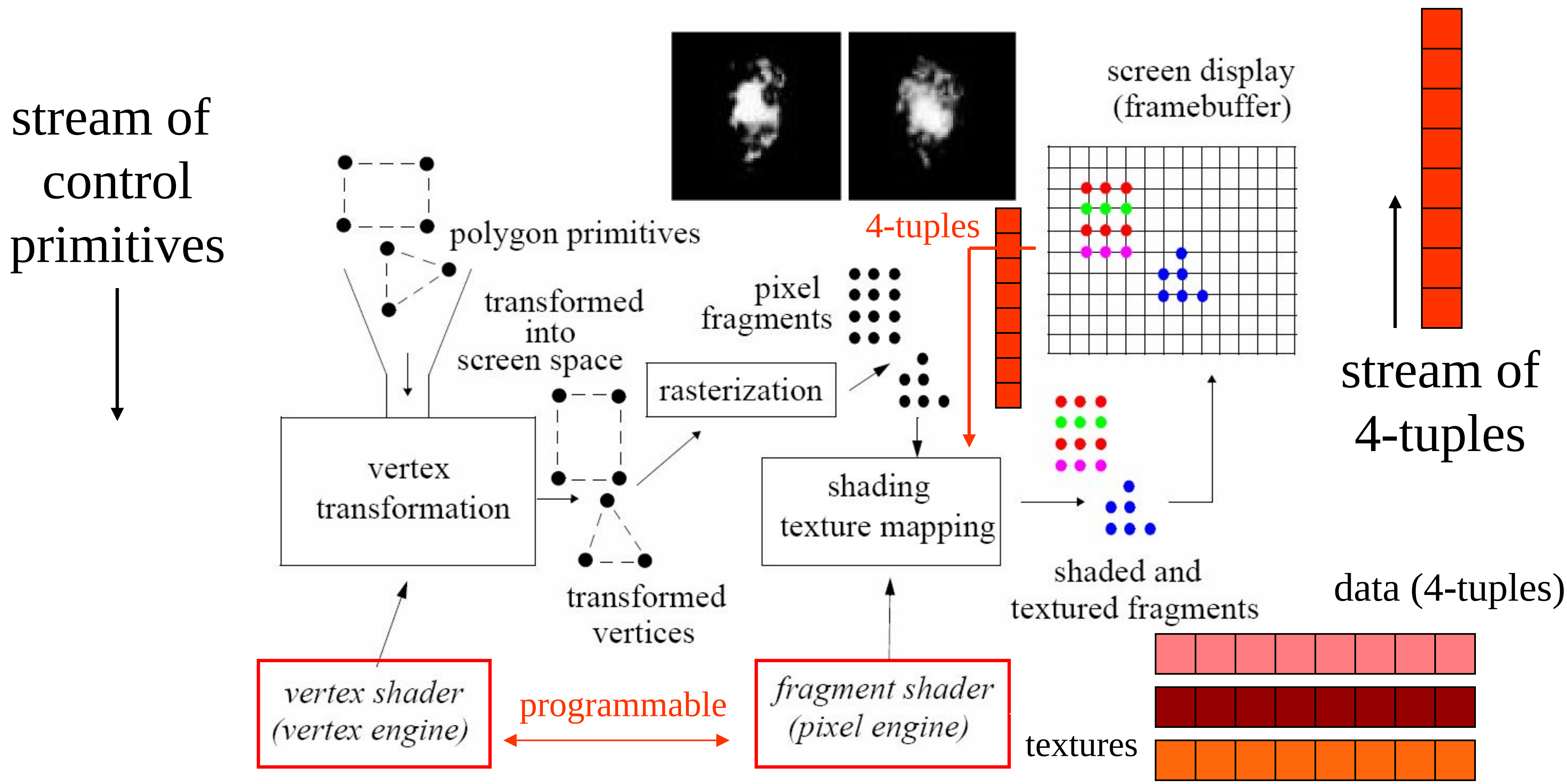
The Graphics Pipeline

Modern, programmable:



The Graphics Pipeline

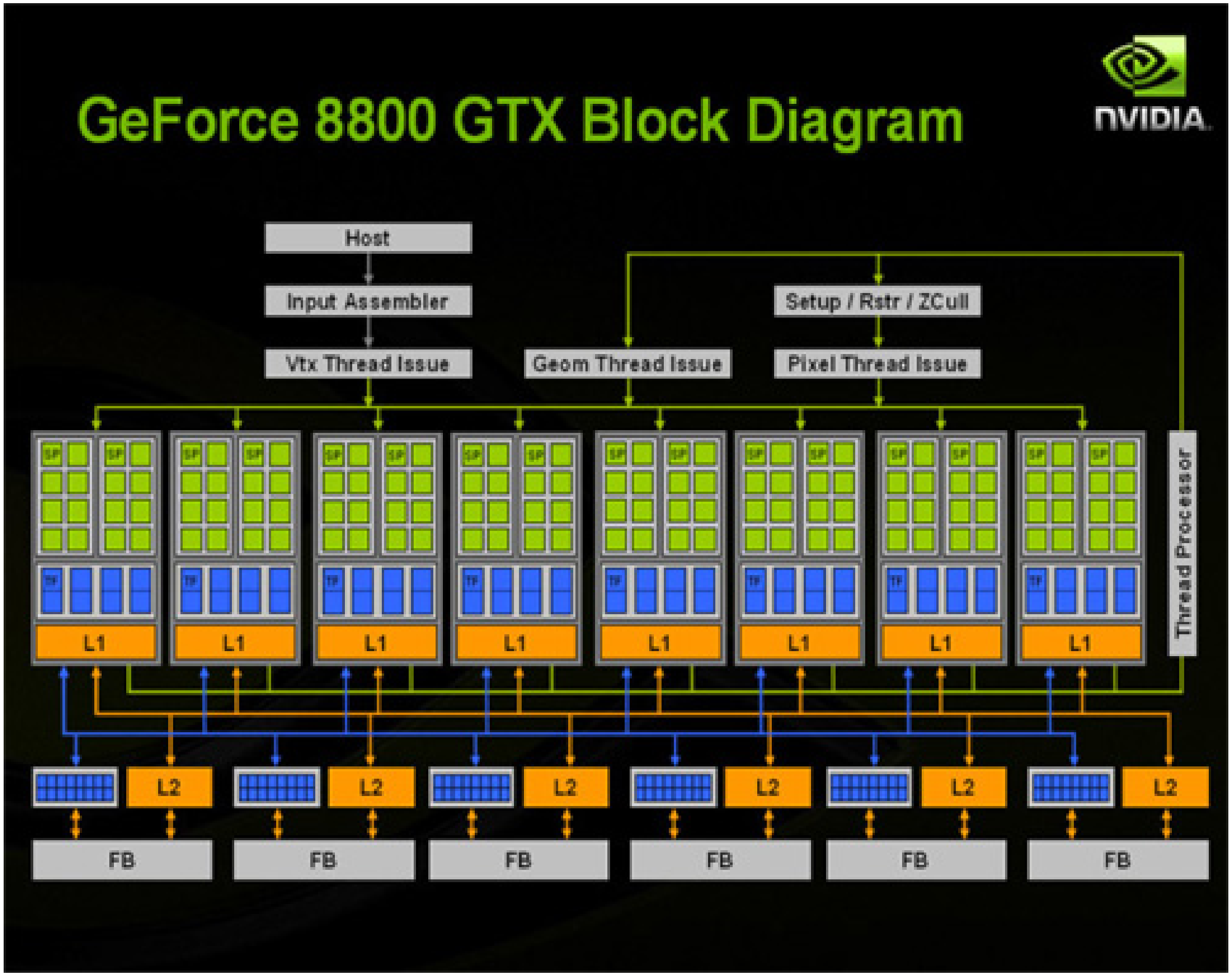
From a computational view:



GPU Vital Specs

	GeForce 7900 GTX	GeForce 8800 GTX
Codename	G71	G80
Release date	3/2006	11/2006
Transistors	278 M (90nm)	681 M (90nm)
Clock speed	650 MHz	1350 MHz
Processors	**24+8 (pixel/vertex)**	**128 (unified)**
Peak pixel fill rate	10.4 Gigapixels/s	36.8 Gigapixels/s
Pk memory bandwidth	51.2 GB/s (256 bit)	86.4 GB/s (384 bit)
Memory	512 MB	768 MB
Peak performance	250 Gigaflops	520 Gigaflops

GPU Block Diagram



high chip real estate for computing (compare 6.5% in Iridium CPU)

128 processors arranged into 8 blocks

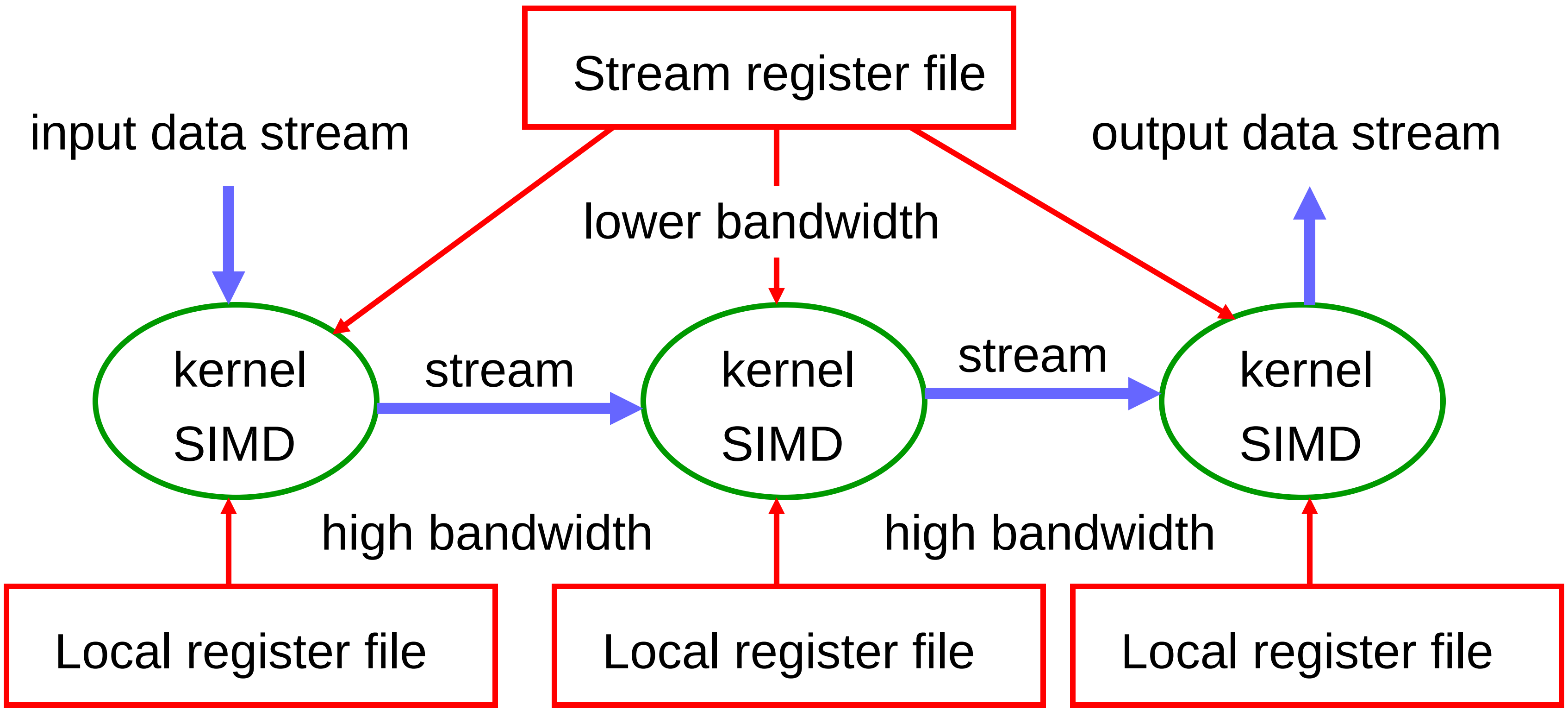
local cache

shared memory

Stream Processing

GPUs are *stream processors* [Kapasi '03]

(with some restrictions) [Venkatasubramanian '03]



Reconstruction Algorithms: Decomposition into Kernels

STONY BROOK Center for Visual Computing

$$P:\ p_i = \sum_{j=0}^{N^3-1} \left(v_j \cdot w_{ij} \right) \qquad B:\ v_j = \sum_{i=0}^{M_\varphi-1} \left(p_i \cdot w_{ij} \right)$$

FBP

$$v_j = \sum_{p_i \in P_{set}} p_i w_{ij\_fdk} = \sum_{p_i \in P_{set}} B(S)$$

S: scanner projections

I: identity projection/volume

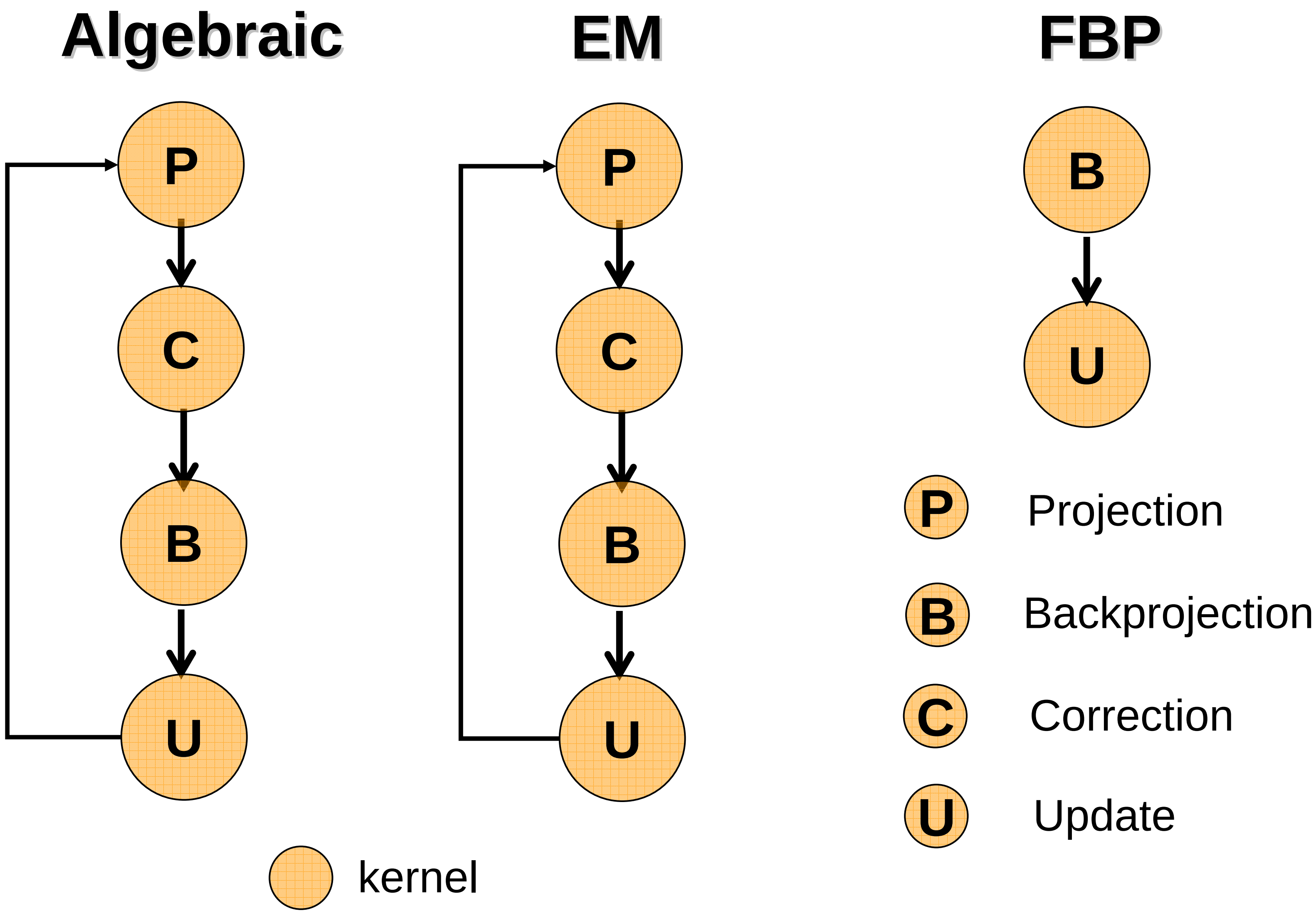
Algebraic

$$v_j = v_j + \frac{\sum_{p_i \in P\varphi} \left(\frac{\lambda \left(p_i - \sum_{l=0}^{N^3-1} v_l \cdot w_{il} \right)}{\sum_{l=0}^{N^3-1} w_{il}} \right) w_{ij}}{\sum_{p_i \in P_\varphi} w_{ij}} = v_j + \frac{B(\lambda \frac{S - P(V)}{P(I)})}{B(I)}$$

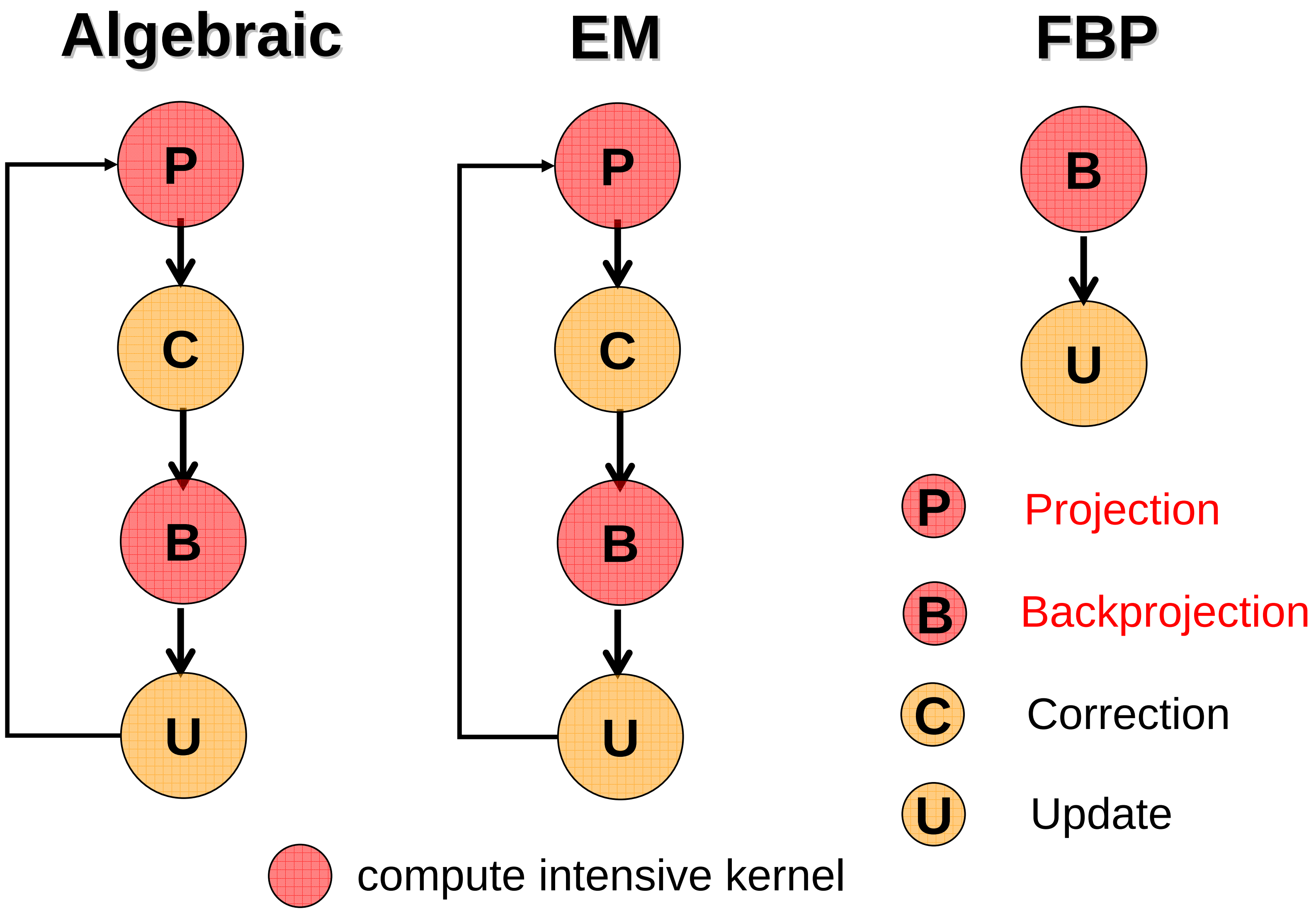
OS-EM

$$v_j = \frac{v_j}{\sum_{p_i \in P_{set}} w_{ij}} \left(\sum_{p_i \in P_{set}} \left(\frac{p_i}{\sum_{l=0}^{N^3-1} v_l \cdot w_{il}} \right) w_{ij} \right) = \frac{v_j}{\sum_{p_i \in P_{set}} B(I)} \left(\sum_{p_i \in P_{set}} B(\frac{S}{P(V)}) \right)$$

Kernel-Centric Reconstruction

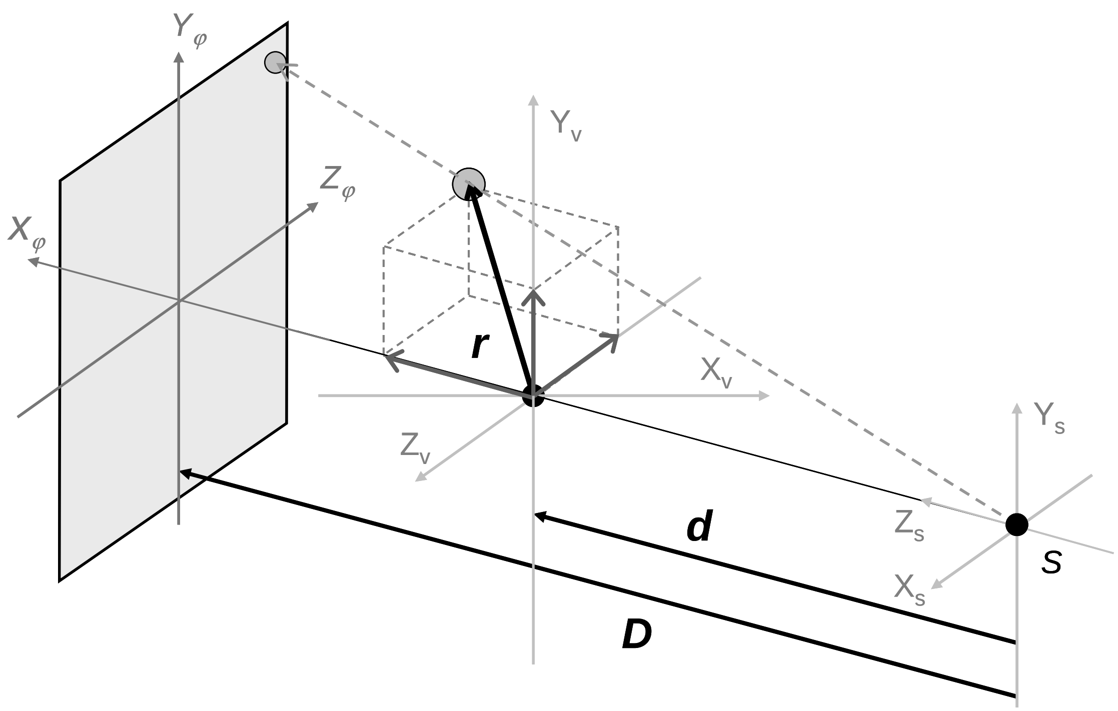


Kernel-Centric Reconstruction

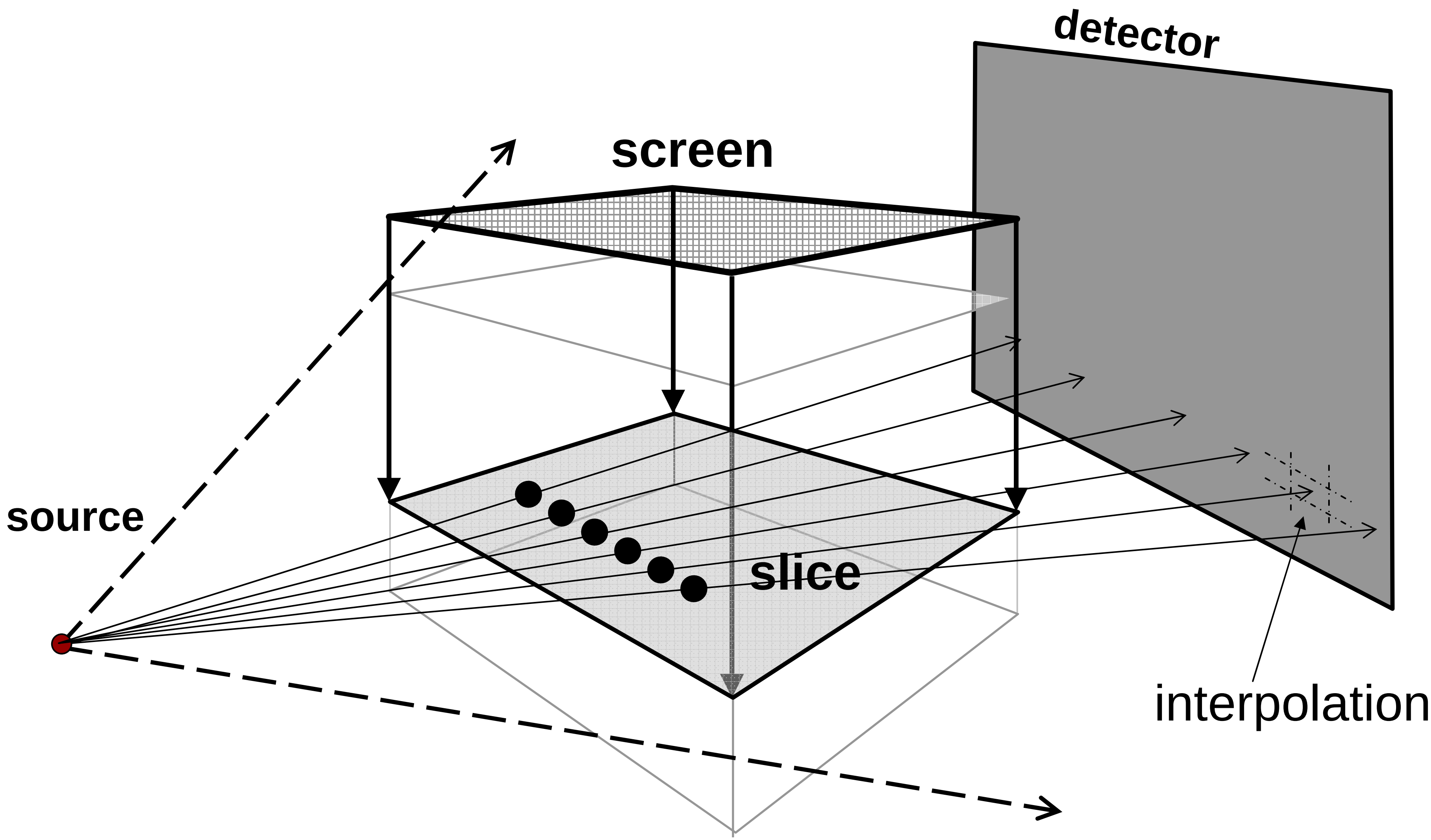


Back-Projection Mapping Via Transformation Matrix

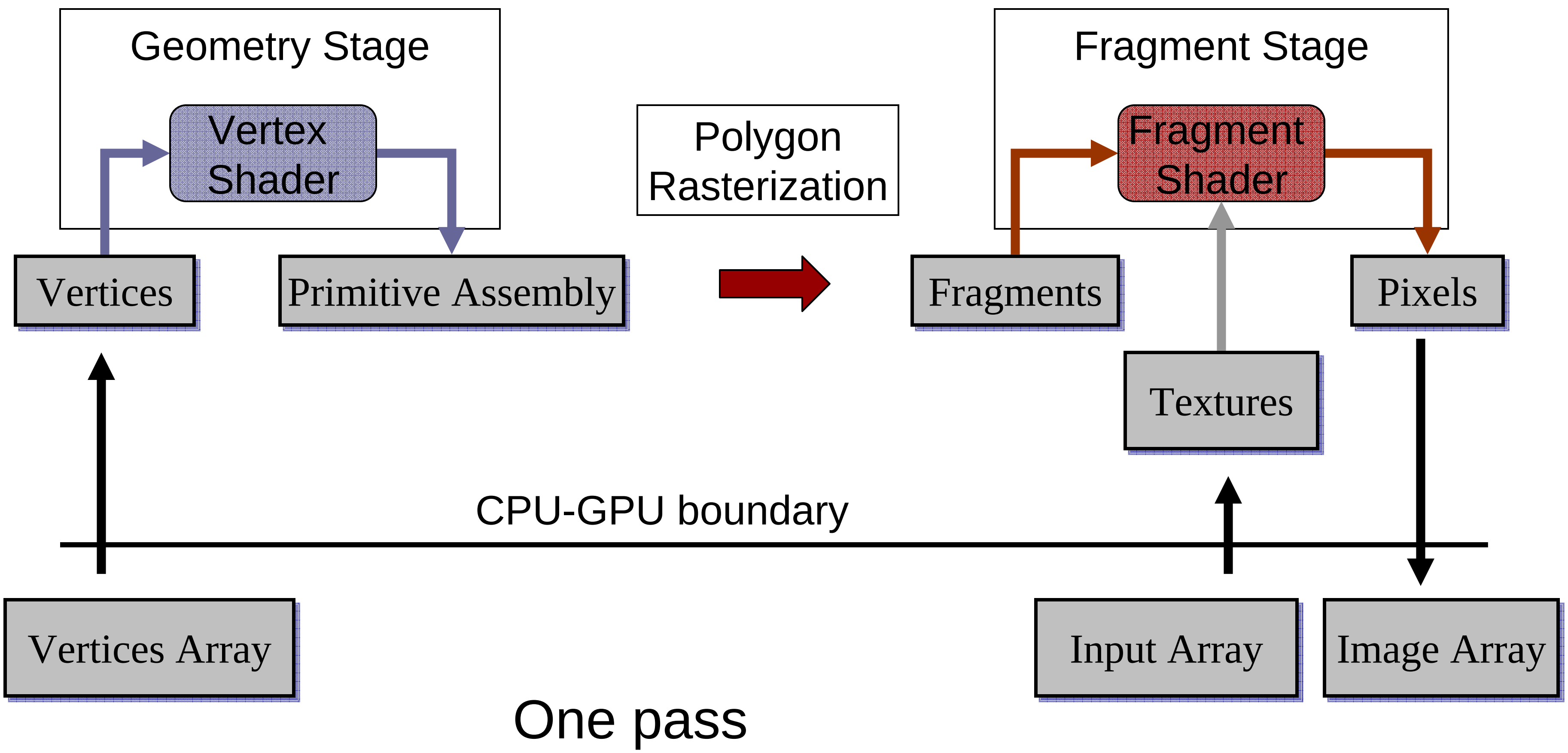
$$
\begin{bmatrix} \frac{w}{2} & 0 & 0 & 0 \\ 0 & \frac{h}{2} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} 1 & 0 & 0 & 1.0 \\ 0 & 1 & 0 & 1.0 \\ 0 & 0 & 1 & 1.0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} \frac{2n}{w} & 0 & 0 & 0 \\ 0 & \frac{2n}{h} & 0 & 0 \\ 0 & 0 & \frac{f+n}{n-f} & \frac{2fn}{n-f} \\ 0 & 0 & -1 & 0 \end{bmatrix}
\begin{bmatrix} u_x & u_y & u_z & -\vec{u}\cdot\vec{s} \\ v_x & v_y & v_z & -\vec{v}\cdot\vec{s} \\ n_x & n_y & n_z & -\vec{n}\cdot\vec{s} \\ 0 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} x_v \\ y_v \\ z_v \\ 1 \end{bmatrix}
$$



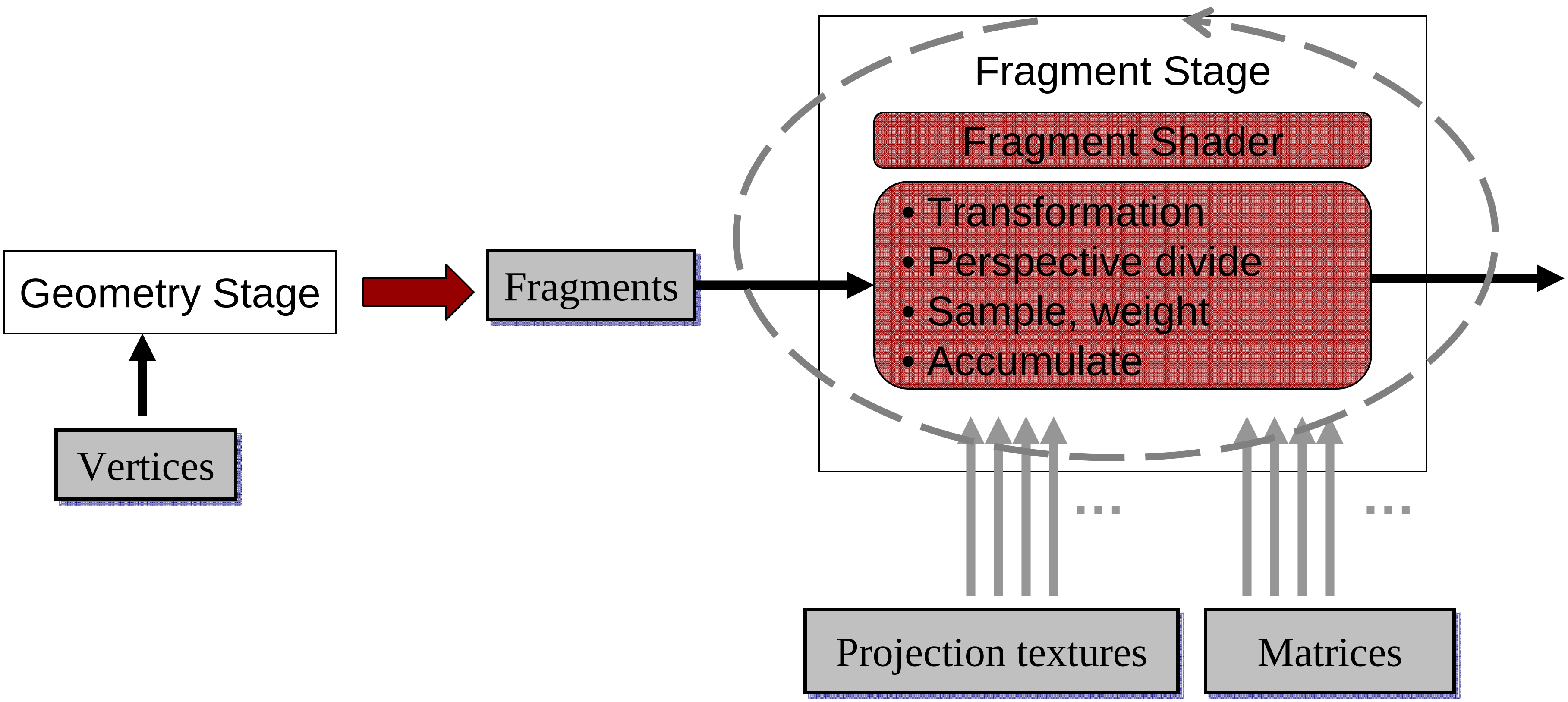
Implementation Via Parallel Fragment Processing



Graphics Pipeline Revisited

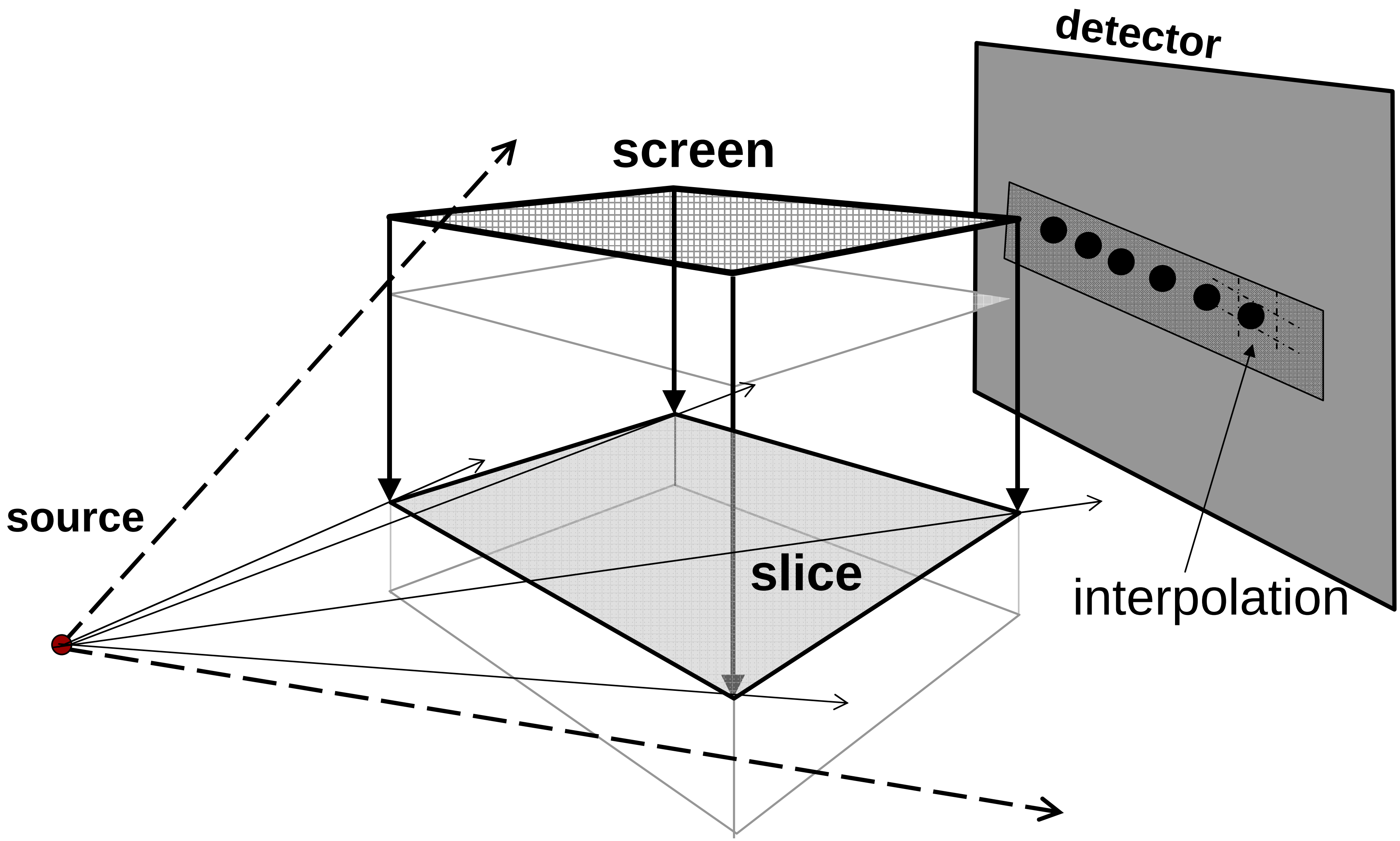


Pipeline 1: The GPU as a Programmable Multi-Processor (MP-GPU)

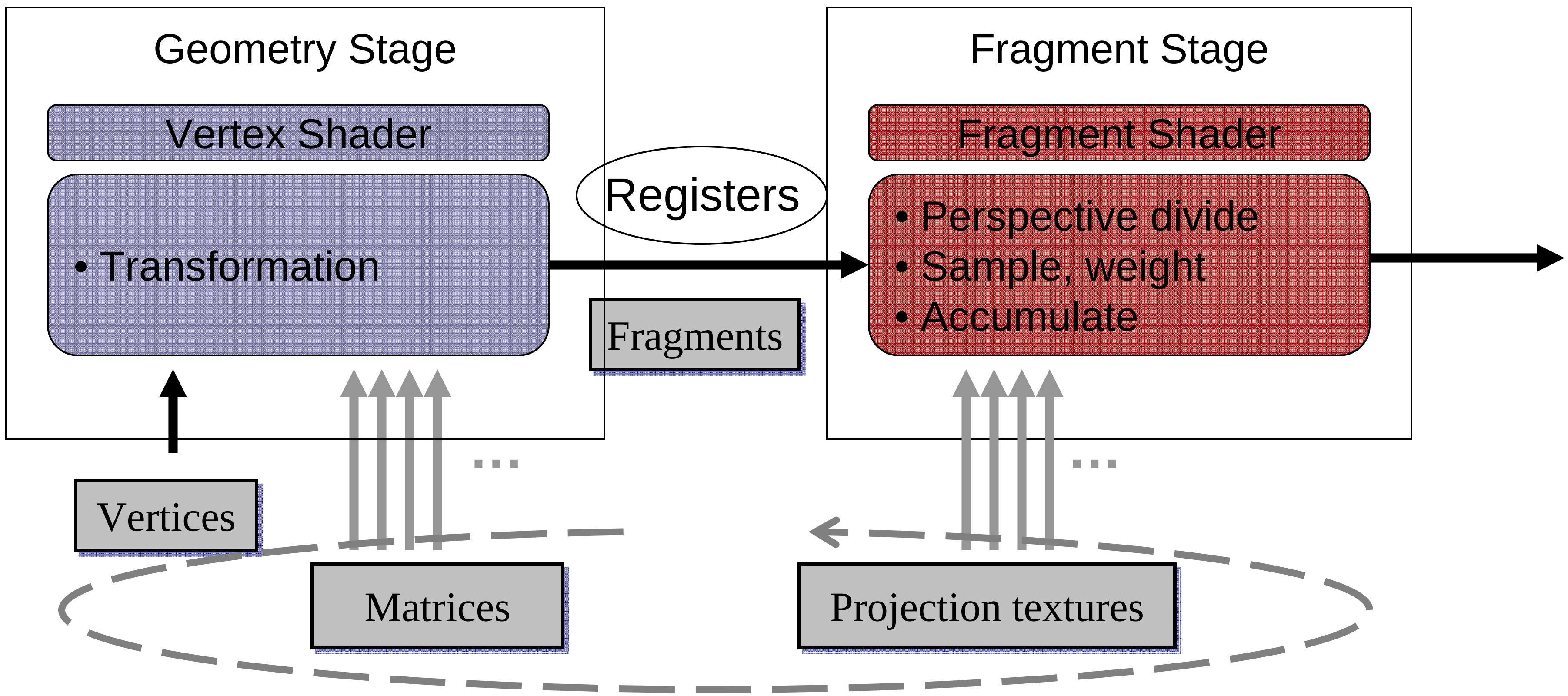


Fragments contain the (x,y,z) voxel coordinates

Implementation Via Parallel Fragment Processing



Pipeline 2: The GPU as a Programmable Graphics Processor (AG-GPU)



Fragments contain the (u,v) detector space coordinates

Graphics Pipeline Benefits

STONY BROOK
Center for Visual Computing

Graphics-aware pipeline (AG-GPU) is considerably faster (~3×) than MP-GPU

- graphics facilities are hardwired!

There are further features that have their origins in graphics and come with GPUs:

- early fragment kill → eliminate fragments based on some condition before they even enter the fragment processor
- hardwired 32-bit floating-point precision linear interpolations, matrix and vector arithmetic (+, -, *), frame-buffer blending and compositing
- RGBA parallelism

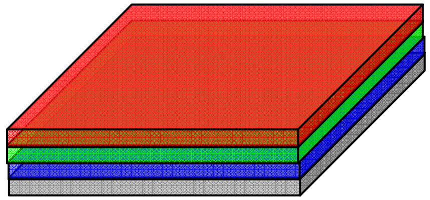
see Xu/Mueller, *Physics Medicine & Biology*, vol. 52, pp. 3405–3419, 2007

RGBA Parallelism

Exploit geometric mapping parallelism

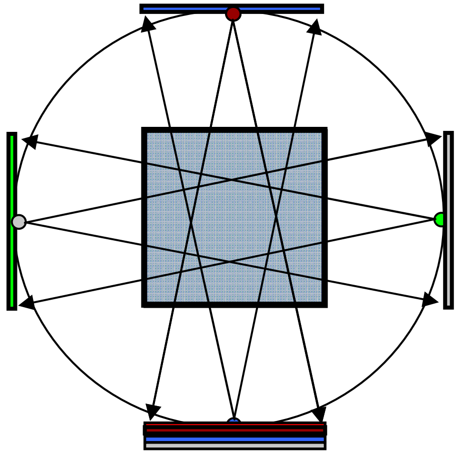
Volume packing

- adjacent 4 volume slices ➔ RGBA



Projection packing

- symmetry in projection layout
- requires all projections beforehand

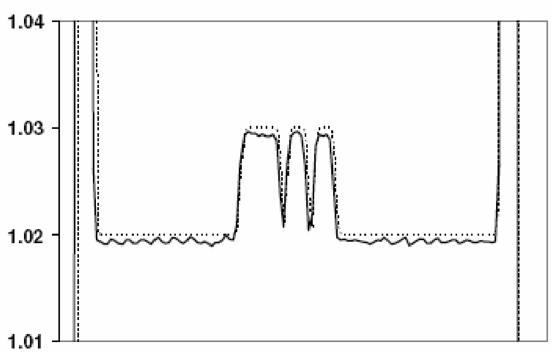


Example:
Feldkamp Cone-Beam Reconstruction

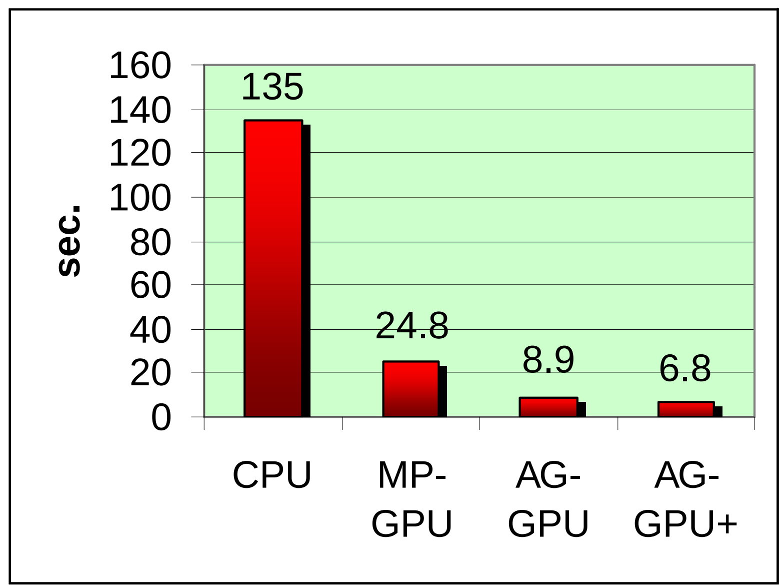
360 projections (1024^2, general position), 512^3 volume

CPU

GPU



tumor profiles

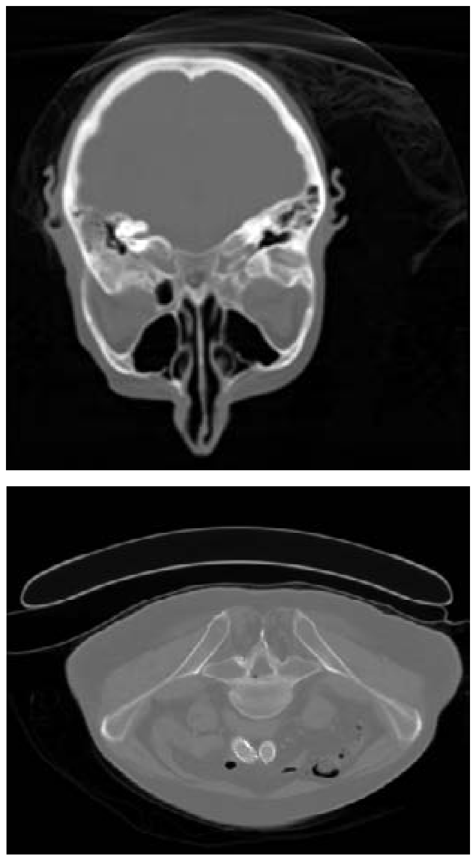


performance

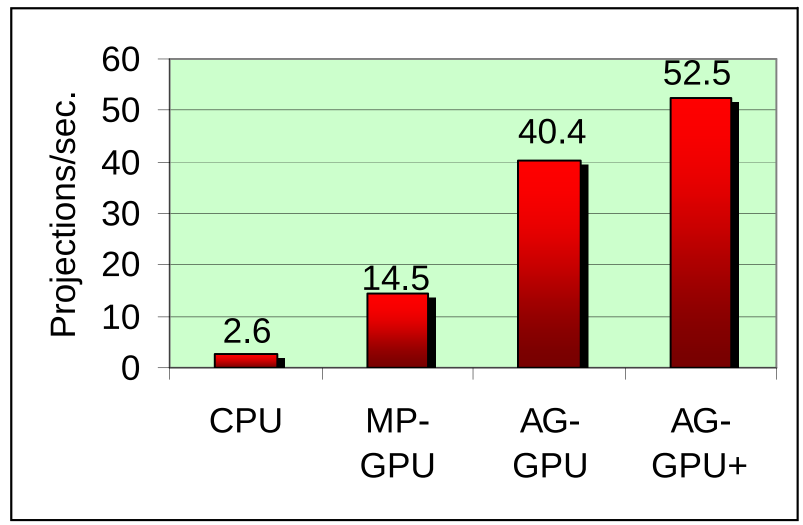
Expressed in Projections/Sec.

360 projections, 512^3 volume

Original



GPU-recon

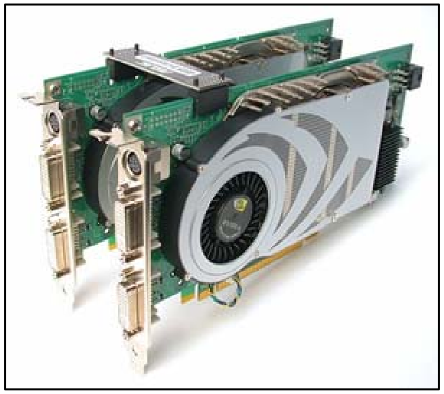


performance in projections/s

GPU Enables Streaming CT

Real-time reconstruction at clinical rates

- reconstruct (consume) incoming (produced) projections without buffering



GPU Enables Visual CT

High reconstruction frame rate enables injection of occasional volume rendering step

Also enables D^2VR: real-time volume visualization directly from projection data

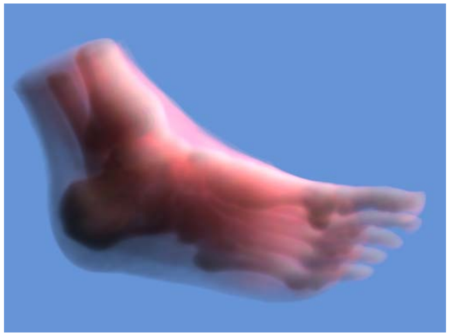
see Xu/Mueller, *Proc. Volume Graphics Workshop*, pp. 23-30, 2006

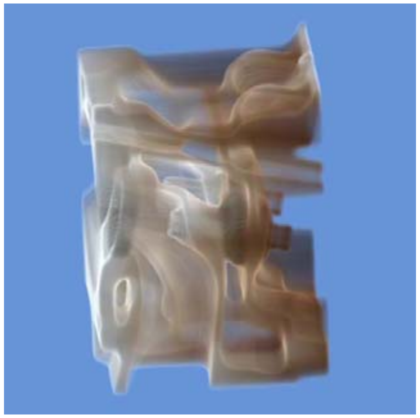
Extensions

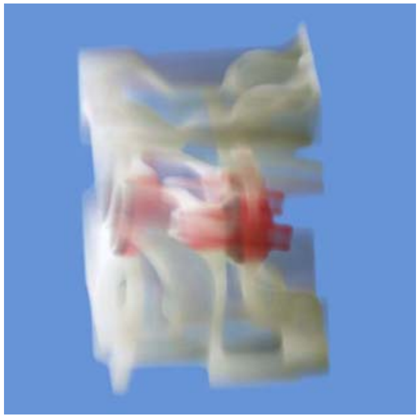
GPUs can not only be used to accelerate straight projection and back projection

They also allow more complex effects to be modeled

- can use relatively simple fragment programs for scatter and attenuation modeling
- but we have recently also implemented more complex scattering models using lattice-based Monte-Carlo techniques







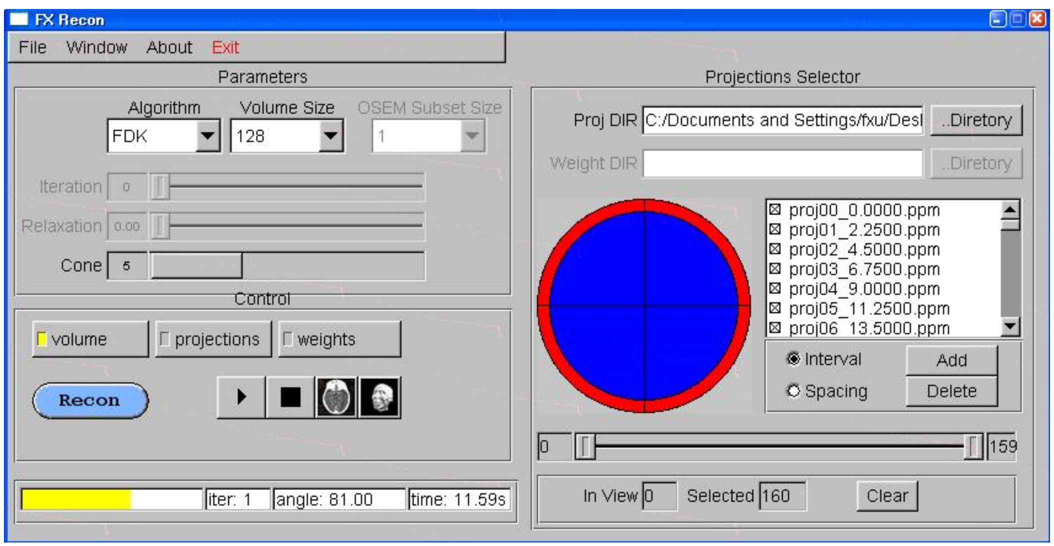
Qiu et al, *Trans. Vis. Comp. Graph*, 2007

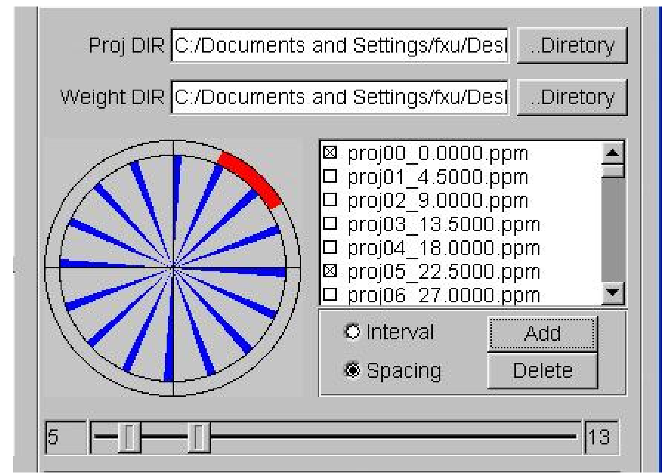
RapidCT Reconstruction Cockpit

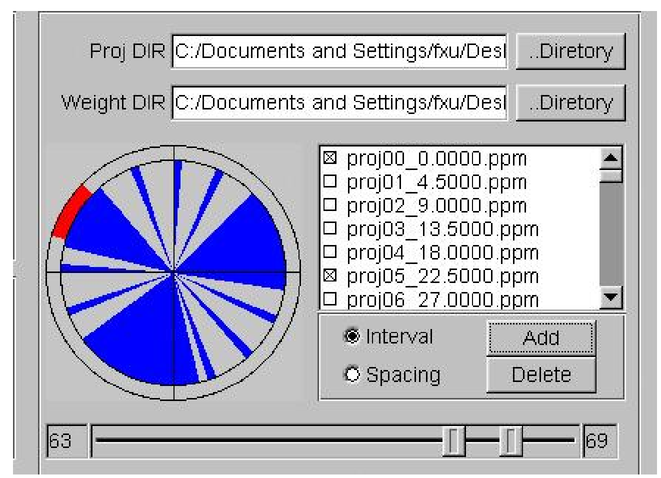
Edit/tune on the fly:

- parameters
- projection sets
- algorithms

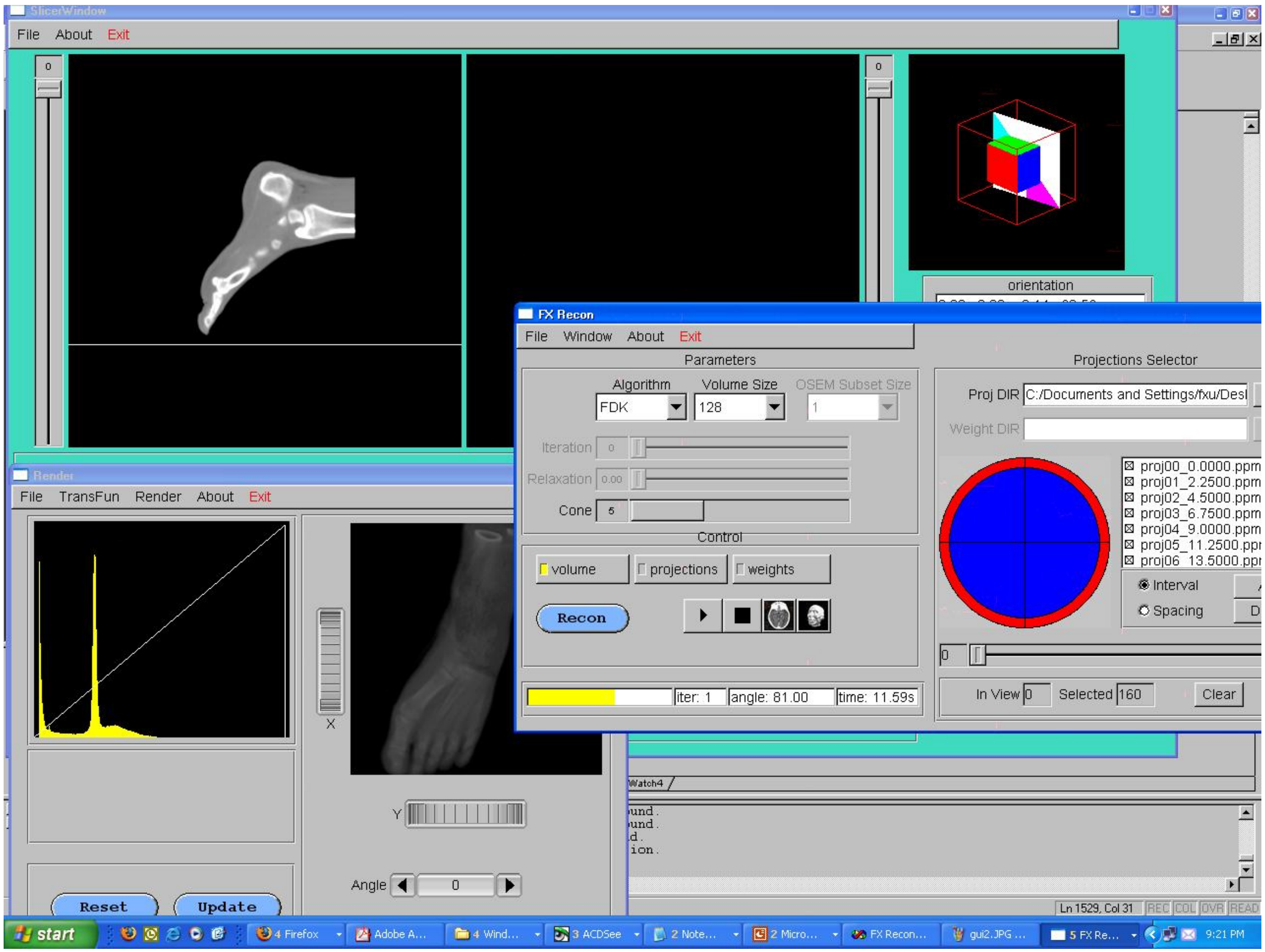
Couple with 2D/3D visualizations







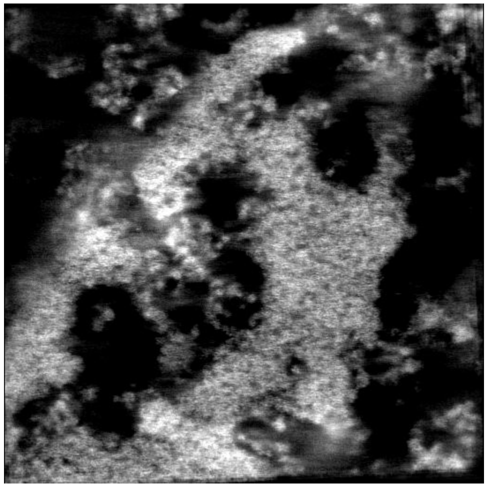
RapidCT Reconstruction Cockpit



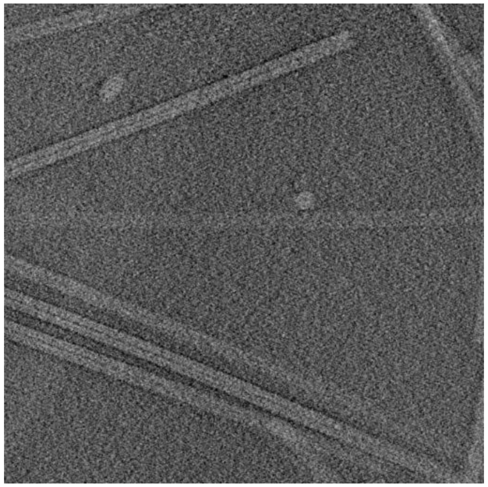
One More Example: Algebraic Reconstruction of TEM Data

GPUs enable iterative reconstruction from large data in Transmission Electron Microscopy (TEM)

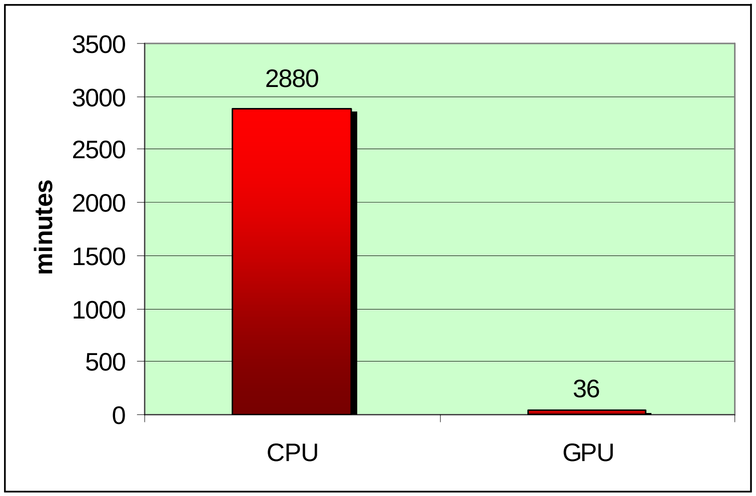
- 70 1024^2 parallel-beam projections, 130° tilt angle,
- reconstructed with SIRT at 50 iterations into a 1024^3 volume
- uses RGBA parallelism and sinogram-centric fragment generation



chromatin



tobacco mosaic virus



performance

Questions?

Funding by NIH, NSF, Keck Foundation