Medical Image Processing for Interventional Applications

Real-time Preprocessing using GPUs

Online Course – Unit 17

Andreas Maier, Jakob Wasza, Frank Schebesch
Pattern Recognition Lab (CS 5)

virtuelle hochschule bayern

FRIEDRICH-ALEXANDER UNIVERSITÄT ERLANGEN-NÜRNBERG
TECHNISCHE FAKULTÄT

Topics

Real-time Preprocessing using GPUs

- Implementation Strategies
- Numerical Issues

Summary

- Take Home Messages
- Further Readings

Motivation: Why Use GPUs?

Challenge: Real-time capable preprocessing pipeline

→ Time constraints in interventional imaging

→ Driving business decisions: “to buy or not to buy”

Solution: GPU based implementations

- High degree of parallelism in the GPU architecture
- Concurrent execution of image processing tasks
- Hardware accelerated interpolation

Figure 1: NVIDIA GTX 680 (left), Quadro K5000 (middle), Tesla K20X (right) (image source: Nvidia)

Hardware Architecture for CPUs and GPUs

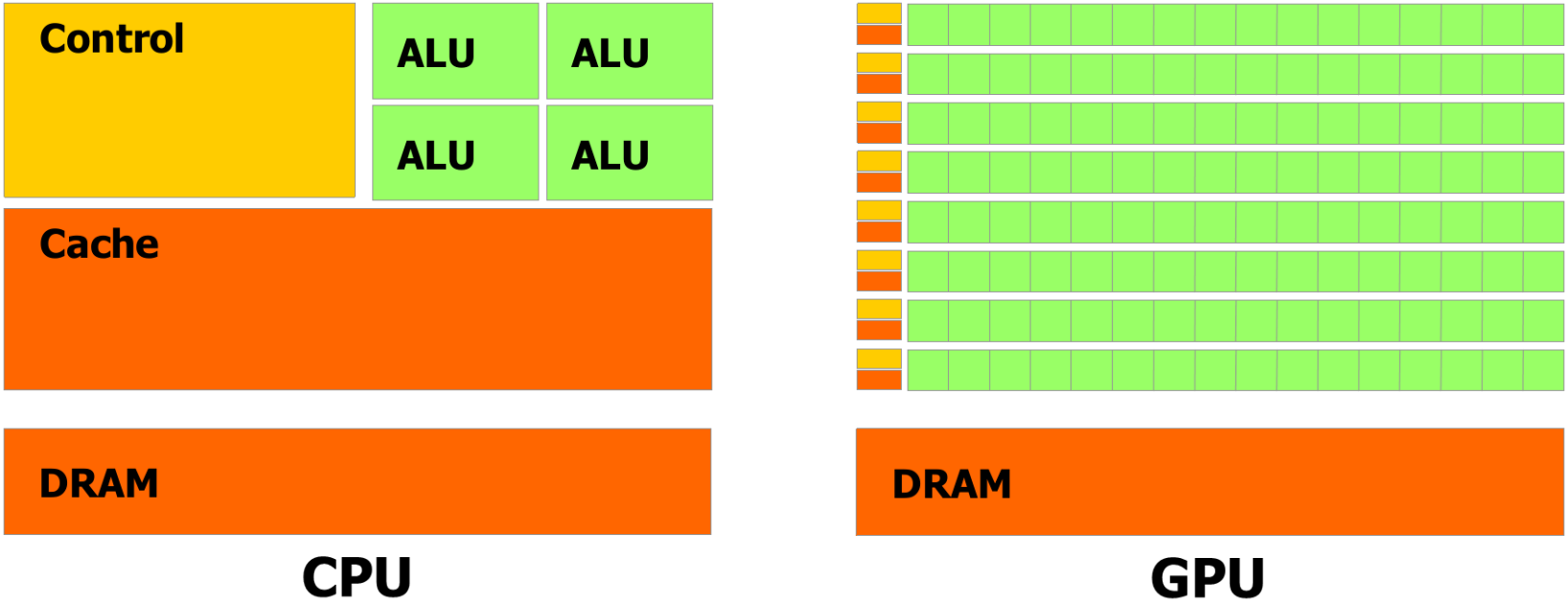

Figure 2: These schematics of CPU (left) and GPU (right) architecture show that the GPU devotes more transistors to data processing.

Application to Preprocessing

- Recall the bilateral filter:

$$f_{\mathrm{BF}}(\boldsymbol{x}) = \frac{\sum_{\boldsymbol{x}' \in \omega_{\boldsymbol{x}}} g(\boldsymbol{x}) c(\boldsymbol{x}, \boldsymbol{x}') s\big(g(\boldsymbol{x}), g(\boldsymbol{x}')\big)}{\sum_{\boldsymbol{x}' \in \omega_{\boldsymbol{x}}} c(\boldsymbol{x}, \boldsymbol{x}') s\big(g(\boldsymbol{x}), g(\boldsymbol{x}')\big)}$$

- Has to be computed for all pixels $\boldsymbol{x}$
  - → Assign a (light-weighted) GPU thread to each pixel
  - → Massive parallelization
- Memory access patterns in $\omega_{\boldsymbol{x}}$ and boundary conditions
  - → Texture cache designed for 2-D spatial locality
  - → Texture clamping
- Re-use components (spatial closeness)

Numerical Issues

- Single-precision floating-point numbers:
  - IEEE 754-2008 standard
  - Seven decimal digits
- Output is different from the mathematically exact result:
  - rounding errors or accuracy,
  - overflow.
- Consequence for guided filtering:
  - In particular: $\mathrm{Cov}(X,Y) \neq \mathrm{E}(XY) - \mathrm{E}(X)\mathrm{E}(Y)$
  - Integral images → exploit the separability of mean filtering: $X * \mathcal{M}_r = \boldsymbol{m}_r^T * (\boldsymbol{m}_r * X), \boldsymbol{m}_r = [1, \dots, 1]$

Real-time Preprocessing Using GPUs

Figure 3: Erroneous covariances using std integral images and mean filtering

Figure 4: Erroneous covariances using separable mean filtering

Real-time Preprocessing Using GPUs

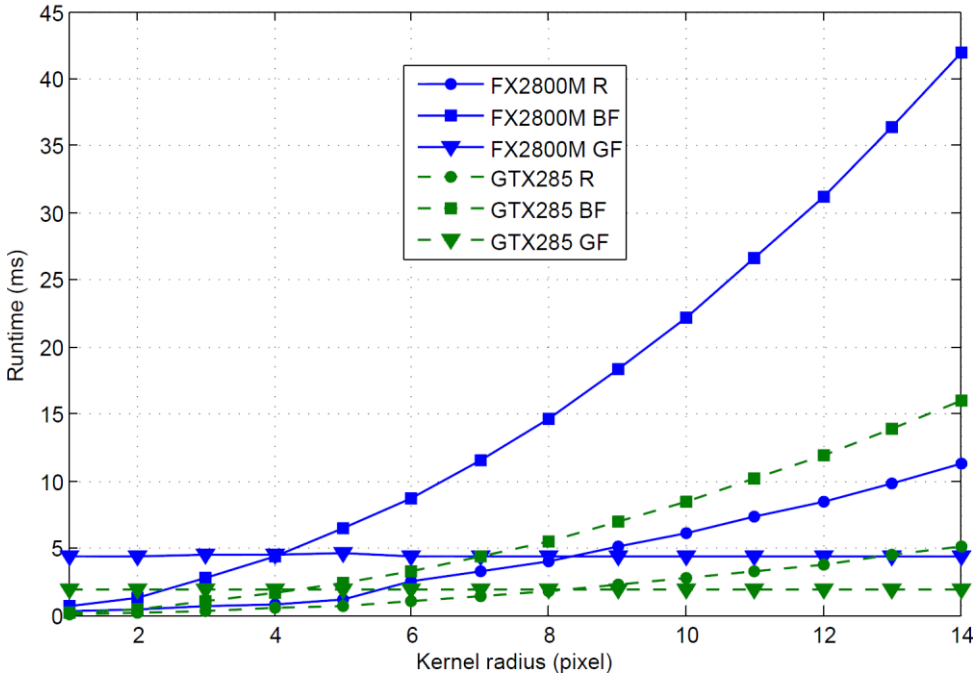

Figure 5: Filter runtimes for VGA resolution (640x480)

Topics

Real-time Preprocessing using GPUs

- Implementation Strategies
- Numerical Issues

Summary

- Take Home Messages
- Further Readings

Take Home Messages

- Preprocessing is a fundamental step which includes the restoration of invalid values and edge-preserving denoising.
- In order to accelerate computation, we discussed algorithmic approaches and the use of GPU hardware.
- The GPU can be a powerful tool when parallelizable preprocessing steps are performed on an image.

Further Readings

Collection of the literature used in the last units on bilateral and guided filtering:

- Hans Knutsson and Carl-Fredrik Westin. “Normalized and Differential Convolution: Methods for Interpolation and Filtering of Incomplete and Uncertain Data”. In: *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, June 1993, pp. 515–523. DOI: 10.1109/CVPR.1993.341081
- Carlo Tomasi and Roberto Manduchi. “Bilateral Filtering for Gray and Color Images”. In: *Sixth International Conference on Computer Vision, 1998. Sponsored by the IEEE Computer Society, January 4-7, 1998, Bombay, India*. IEEE, Jan. 1998, pp. 839–846. DOI: 10.1109/ICCV.1998.710815
- Kaiming He, Jian Sun, and Xiaoou Tang. “Guided Image Filtering”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.6 (June 2013), pp. 1397–1409. DOI: 10.1109/TPAMI.2012.213
- Paul Viola and Michael Jones. “Rapid Object Detection Using a Boosted Cascade of Simple Features”. In: *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*. Vol. 1. IEEE, Dec. 2001, pp. I-511–I-518. DOI: 10.1109/CVPR.2001.990517