Large Histogram Computation for Normalized Mutual Information on GPU

Supercomputing 2016, Salt Lake City, UTAH

Sophie Voisin
voisins@ornl.gov

Devin White
whiteda1@ornl.gov

Jeremy Archuleta
archuletajs@ornl.gov

Introduction

This poster presents two GPU implementations of normalized mutual information (NMI) for a two-step image registration used in version 1.0 of the Photogrammetric Registration of Imagery from Manned and Unmanned Systems (PRIMUS) pipeline [1]. This fully-automated application aims to produce precise geolocation of sensor-agnostic imagery, building block for accurate geographic applications and data analysis.

Although NMI is of common use to register images from different modalities [2], GPU implementations usually focus on single coefficient computation leveraging entropy approximation calculus or generating histograms with a limited number of bins [3-5]. For our use case, we aim to generate the full joint-histogram of 65536 bins to mitigate compression that occurs when reducing the image-intensity range from 16-bit to 8-bit. Our preliminary approaches that target the NVidia Tesla K80, not only leverage the global memory available on one of the GPU but also the GPU queue manager and the specificity of our problem.

Normalized Mutual Information

Similarity measurement between two images from different modalities.

$$NMI(I_1, I_2) = \frac{(H(I_1) + H(I_1))}{H(I_1, I_2)}$$

$$H(I) = -\sum_i^n P(i) log_2(P(i))$$

In PRIMUS, NMI is used for coarse and refined alignment. However, different NMI algorithms have been implemented to fit the different characteristics of each alignment.

Discussion

This preliminary work highlights the need for multiple NMI algorithms depending on the specifications of the imagery and their use cases. For this work, our target GPU was only one of the two GPUs present on the NVidia Tesla K80 but we plan to expand our research to leverage the additional computational power on the full device. A significant amount of research and development is still needed to optimize NMI algorithms for imagery applications where either the images themselves and/or the solution space are larger than the memory and computation resources on the GPU.

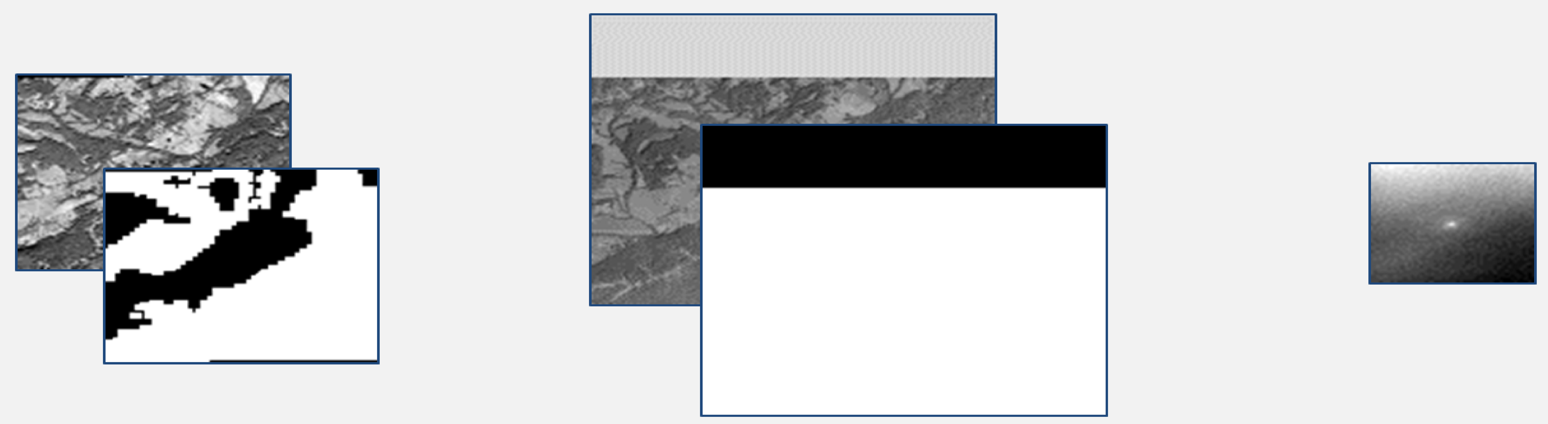
Acknowledgement:

This manuscript has been authored by UT-Battelle, LLC under Contract No. DE-AC05-00OR22725 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paidup, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. The Department of Energy will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan(http://energy.gov/downloads/doe-public-access-plan).

Coarse Alignment

Dense Joint Histogram for Full Image Matching:

- *Problematic:*
  - Template matching on low resolution images.
    - ~400x smaller than second step
    - source image: 500x100 pixels
    - control image: 1200x300 pixels
  - Rigid deformation (translation)
  - Exhaustive search of the solution space.
    - NMI coefficients: 140,000
  - Mask is used to indicate valid pixels



Source image and mask: N_SxM_S pixels

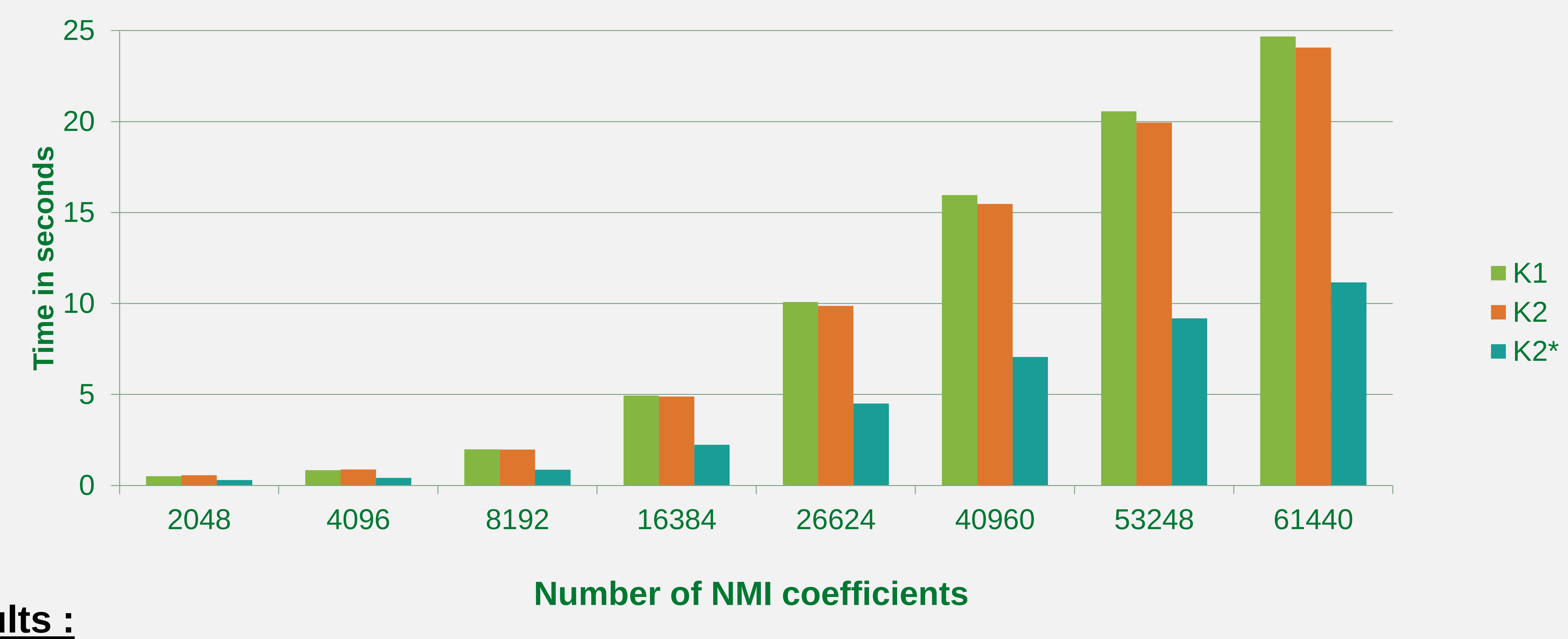
Control image and mask: N_CxM_C pixels

Solution space: (N_C-N_S)x(M_C-M_S) *NMI* coefficients

Approach:

- Hardware
  - 1 GPU – NVidia Tesla K80 (1/2 card)
- Kernel organization:
  - 1 kernel computes all the n NMI coefficients
    $[n/128]$ blocks
    128 threads per blocks
- 1 NMI coefficient computation per thread
  - No synchronization
  - No atomic operation
- Global memory to store all the joint-histograms
- GPU queue manager
- "if" statement to reduce memory access
- Minimize divergence from conditional statement

K1
Consider **all the pixels** to create the joint-histogram
K2
"if" statement to create the joint-histogram



Results :

- *Performance:*
  - ~**25 seconds** for ~**60K NMI coefficients**
  - similar timings for K1 and K2 with **100% valid data**
  - K2* with mask containing **50% of valid** data takes **½ time**.
    - less: memory accesses \ computations - indices to joint-histogram entries \ increment operations
    - low likelihood of divergence for if statement - on-valid pixels close from each other (same warp)
  - linear relation between time and NMI coefficients computation

Refine Alignment

Sparse Joint Histogram for Keypoint Descriptor Matching:

- *Problematic:*
  - Keypoint matching.
  - High resolution sub-images as descriptors.
    - 11x11 descriptor = source image.
    - 73x73 window search = control image.
  - Exhaustive search of the solution space.
  - Rigid deformation: translation with no scaling effects.
  - Solution space: 63x63 NMI coefficients per keypoint
  - Individual keypoint matching problem is not at scale to "fill" the GPU.

Original source image
✚ keypoints

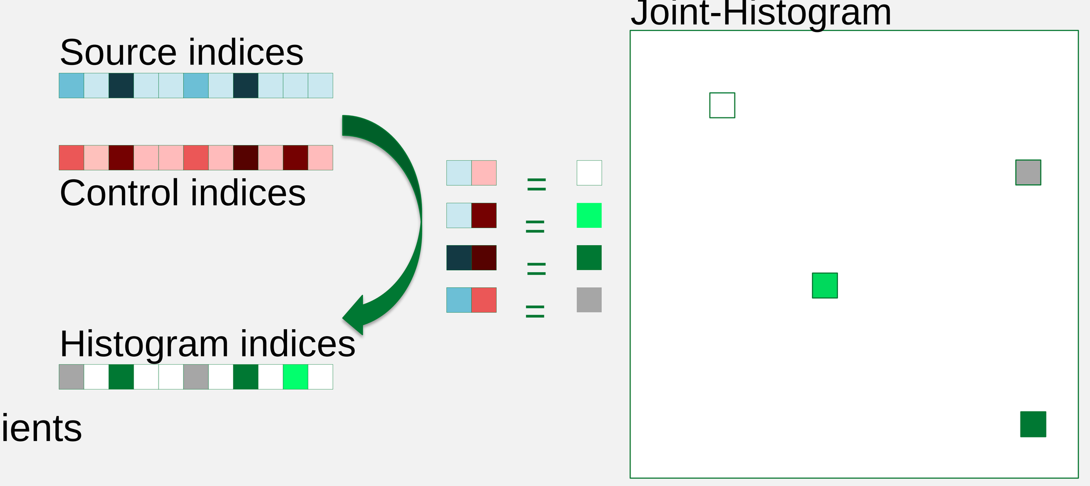
Descriptors:
11x11 intensity values

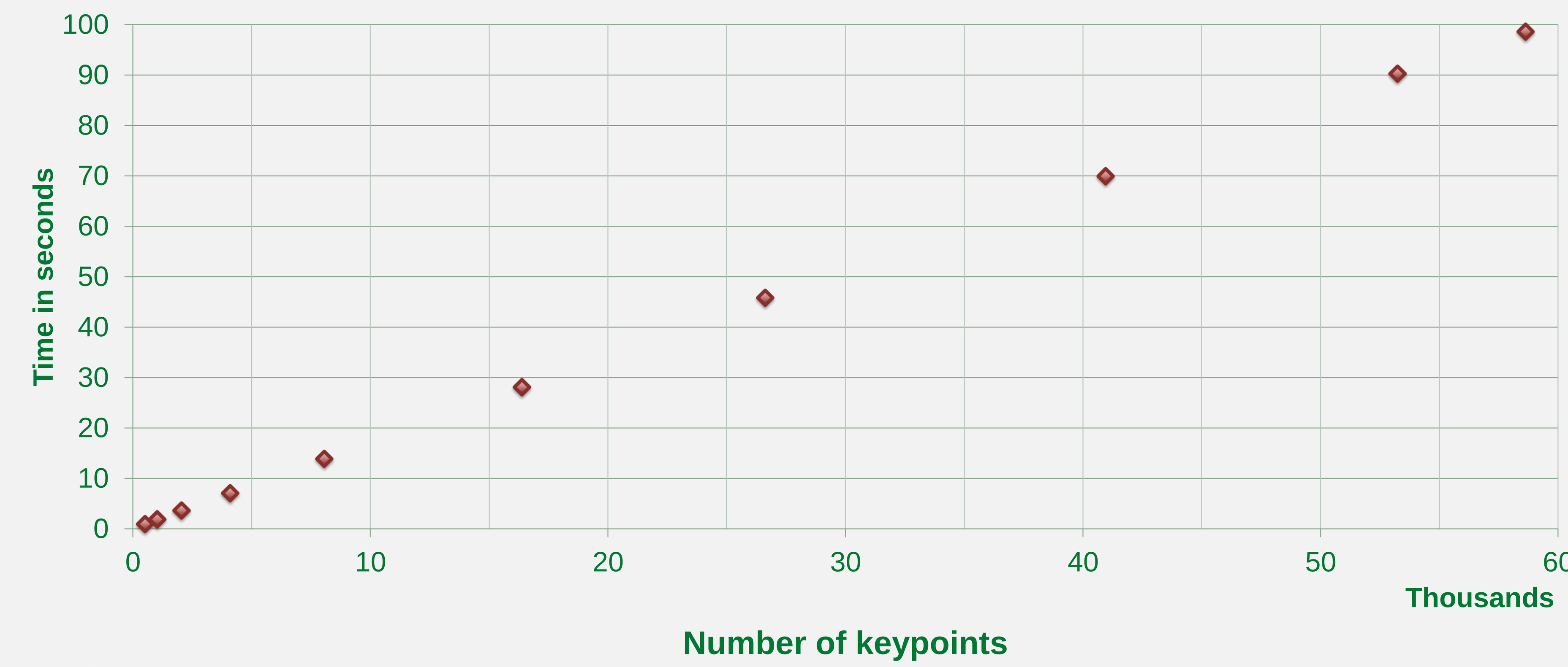
Search windows:
73x73 pixels

Solution spaces:
63x63 *NMI* coefficients

Approach:

- Hardware
  - 1 GPU – NVidia Tesla K80 (1/2 card)
- Kernel organization:
  - 1 kernel computes all NMI coefficients
  - n blocks – 1 per keypoint
    - 1block computes 3969 NMI coefficients
    - 64 threads per block (1idle)
    - 1 thread computes 63 NMI coefficients
- Global memory to store the joint-histograms
- Shared memory to store the intermediate coefficients
- GPU queue manager
- List to locally store joint-histogram entries
  - At most 121 distinct values from the 11x11 descriptor





Results:

- ~**100 seconds** is reported to go over ~**60K keypoints** i.e. computing more than **230M NMI coefficients**
- linear relation between time and keypoints

[1] Devin A. White and Chris R. Davis, A fully-automated high performance image registration workflow to support precision geolocation for imagery collected by airborne and spaceborne sensors, Proceeding on GeoComputation, Dallas, TX, USA, 2015.
[2] F. Maes, D. Loeckx, D. Vandermeulen, and P. Suetens, Image registration using mutual information, N. Paragios et al. (eds.), Handbook of Biomedical Imaging: Methodologies and Clinical Research, ©Springer Science+Business Media New York 2015
[3] Cedric Nugteren, Gert-Jan van den Braak, Henk Corporaal and Bart Mesman, High Performance Predictable Histogramming on GPUs: Exploring and Evaluating Algorithm Trade-offs, GPGPU-4, Newport Beach, CA, USA, March 2011.
[4] R. Shams andR. A. Kennedy, Efficient histogram algorithms for NVIDIA CUDA compatible devices, Proceeding of International Conference on Signal Processing and Communications Systems, ICSPCS, 17-19 December 2007.
[5] Yuping Lin and Gerard Medioni, Mutual information computation and maximization using GPU, IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008. CVPRW '08, Anchorage, Alaska, USA, 23-28 June 2008.