SCALING AUTOMATIC VECTOR DATA ALIGNMENT TO SATELLITE IMAGERY

Abhishek Potnis, Dalton Lunga, Philipe Dias, Lexie Yang, Jacob Arndt, Jordan Bowman

Geospatial Science and Human Security Division, Oak Ridge National Laboratory, Oak Ridge, USA

ABSTRACT

Given the tremendous volume of accessible Earth Observation (EO) data, there is a need to develop scalable Geospatial Artificial Intelligence (GeoAI) solutions for time-sensitive applications. Scalability in this context refers to rapidly processing large-scale EO data using high performance computing resources. Accurate mapping of the built environment from remote sensing (RS) imagery has been one of the crucial components in GeoAI workflows for a wide spectrum of humanitarian applications. Derived vector data of built environment is often leveraged for disaster preparedness and response activities. However, factors such as differences in ortho-rectification, atmospheric conditions and human error, results in spatial misalignment between vector data and the timely available RS imagery. Model training for downstream tasks such as object detection, change analysis, etc., is negatively impacted due to such spatial misalignment. Although there has been progress towards automatic alignment of vector data, the lack of scalability remains an open research challenge. This paper proposes to leverage parallel computing to optimize an automatic vector data alignment workflow. It further employs CPU-level multi-core parallelism for improving the performance of the workflow for scalable built environment mapping. We report observations and discuss findings from the preliminary experiments performed on the Summit Supercomputer.

***Index Terms*—** scalable, geospatial, vector data alignment, raster, remote sensing imagery

1. INTRODUCTION

With advancements in remote sensing technology, there has been an avalanche of Earth Observation (EO) data being generated and stored. Due to its volume, velocity and veracity (the 3 V's), it aptly fits into the Big Data paradigm. Thus this entails development of novel systems and methodologies for effective exploration and exploitation of such information-rich geospatial data. In tandem, there has been tremendous progress in development of high-compute systems. However, it is not feasible to subscribe to the idea of unlimited storage and compute. Rapid analysis of incoming remote sensing data to generate actionable insights becomes extremely crucial for time-critical situations, such as rapid disaster response to effectively manage and mitigate damage to life and property. Thus there is a dire need to develop scalable, accelerated and optimized GeoAI workflows for effective resource utilization of available hardware resources.

Accurate mapping of the built environment finds utility in numerous humanitarian applications including urban sprawl analysis, urban planning and also disaster monitoring and management. Building damage assessment greatly benefits from accurate mapping of buildings before and after the event. For example, post an earthquake when the area on the ground is inaccessible, remote sensing platforms such as drones or airplanes form an apt choice to capture data and make sense of the events that have transpired on the ground. In the event of an earthquake, rapid analysis of the pre- and post-event remote sensing imagery can help detect affected buildings including identifying degree of damage sustained. However, the vector data over the built up areas, as derived from a previously captured remote sensing image might not always align spatially with the most recently captured imagery. During such scenarios, it becomes crucial to first spatially align the vector data with the most recent acquired imagery and then proceed with applying geospatial analytic workflows.

2. RELATED WORK

There exist quite a few approaches proposed in recent literature towards performing automatic vector data alignment with remote sensing imagery. [2] proposes to leverage difference in the standard deviation of road and background pixels for alignment of road segments in the vector data. The authors in [3] seek to utilize line segment and corner detection in optical images for geo-registration of vector data. The research described in [4] proposes to use the graph based GrabCut algorithm for foreground extraction, followed by shift-checking in eight directions, shift and distance determination. The authors in [5] pose the problem of spatial misalignment of vector

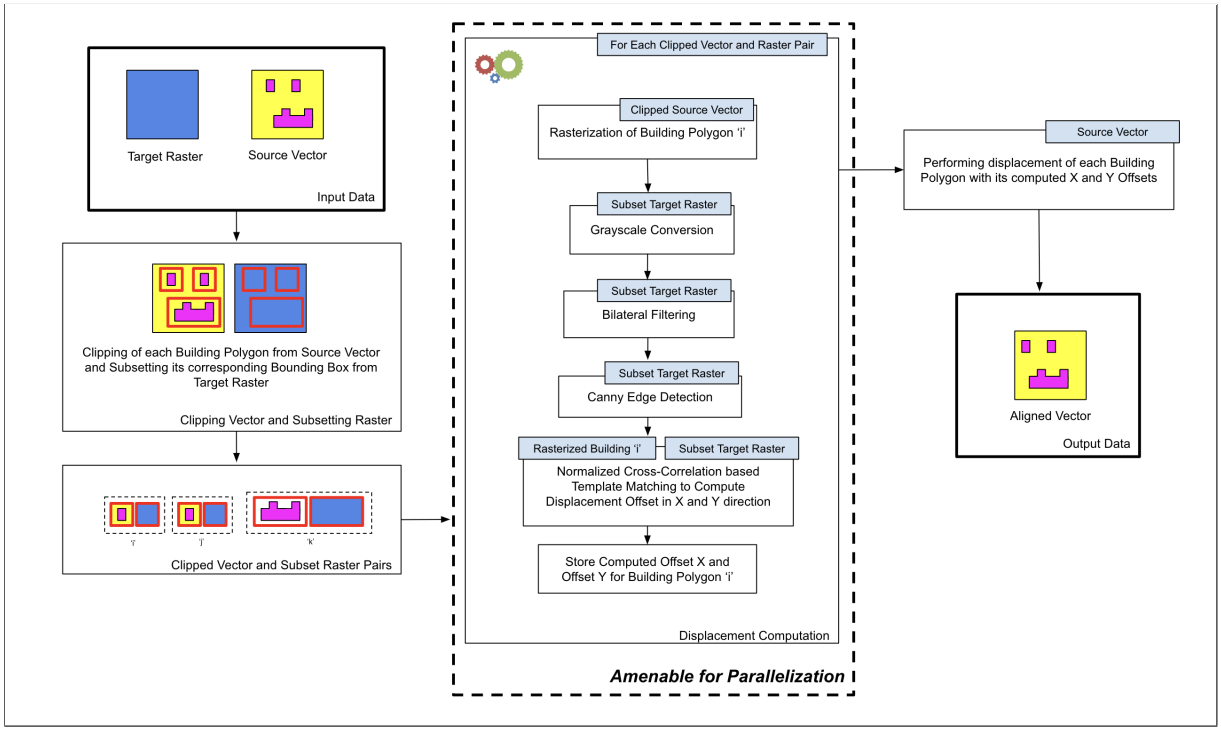

Fig. 1. Workflow of Automatic Vector Data Alignment with Remote Sensing Imagery as proposed in [1]; Dotted Box depicting Displacement Computation - performed independently for each Clipped Vector and Raster Pair, thus being amenable for Parallelization at Scale

data in the paradigm of reinforcement learning, maximizing reward based on the color principle in cartography. However, this approach is geared towards alignment of vector data with geo-referenced historical raster maps and not remote sensing imagery.

In [1], the authors present a novel workflow for alignment of vector data that utilizes bilateral filtering[6] and Canny edge detection[7] followed by normalized cross-correlation-based template matching to compute the displacement offsets in x and y directions for spatial alignment of buildings. Considering the efficacy of this approach and given that it seeks to address the issue of spatial misalignment for geospatial vector data, it has been selected as the candidate approach for this paper for studying, analyzing and optimizing its performance for processing large volumes of remote sensing data. Figure 1 illustrates the workflow for automatic vector data alignment with remote sensing imagery.

Our contributions in this work are two-fold - (1) We improve the existing automatic vector data alignment workflow by leveraging parallel computing using Dask. We document and discuss on the performance improvement of the parallel implementation as compared to its serial counterpart; and (2) We demonstrate the scalability of the improved workflow by deploying the parallel implementation on a high-compute node of the Summit supercomputer and leveraging CPU-level multi-core parallelism. We further perform preliminary experiments towards documenting the performance impact for the improved workflow by increasing the number of CPU cores used, in the high-compute node of Summit.

3. METHODOLOGY

Scaling strategies in computing are typically geared towards accelerating and optimizing workflows to handle increasingly large volumes of data and ensuring efficient resource utilization. In context to accelerating workflows, the parallel computing paradigm refers to identifying independent components in a workflow that can be executed in parallel to minimize the end-to-end execution time. The authors in [8] discuss the relevance of parallel computing in context to geospatial big data. In that regard, we identify the *displacement computation* component of the workflow as depicted by dotted box in Figure 1, to be independent for each clipped vector and subset raster pair, which brings forth an excellent opportunity to leverage parallelization at scale and accelerate the workflow.

Dask [9] is an open-source Python library for parallel computing. It enables to scale a Python code from leveraging single CPU multiple core-level parallelism to multi-node level parallelism in distributed clusters. The *Dask Delayed* construct of Dask was used to facilitate parallelization of the *displacement computation* component of the workflow.

Specifically, the code was refactored to define a function *calcShiftForFeature* with the target raster and a polygon vector as its parameters. For each pair of polygon vector and its corresponding clipped raster, this function computed the displacement x and y values. This function was decorated with *Dask Delayed*, explicitly conveying independent computation using a lazy evaluation strategy. Figure 2 depicts the task graph constructed by Dask for lazy evaluation of the *calcShiftForFeature* function.

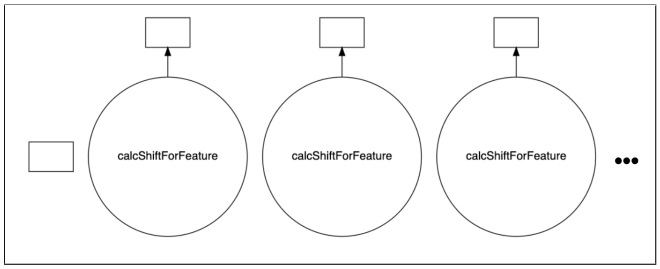

Fig. 2. Task Graph constructed using Dask that depicts lazy evaluation of independent 'calcShiftForFeature' function nodes for each of the clipped vector and corresponding raster pairs

3.1. Experimental Setup

3.1.1. Dataset

This research builds over the previous work by [1], thus a subset of the Microsoft Buildings Dataset [10] over the city of Sioux Falls, South Dakota was selected. The clipped vector data consists of $61,520$ buildings as polygon features with the shape file occupying a disk space of $26.21MB$. The raster selected for this study is a very high resolution (VHR) multi-spectral image tile of size 31502×25523 pixels with a ground sampling distance (GSD) of 0.5 meter, occupying a disk space of $5.99GB$.

3.1.2. High Performance Computing Platform

The Summit supercomputer at Oak Ridge National Laboratory was chosen as the High Performance Computing platform for performing the scalability experiments as a part of this study. This research focused on employing CPU-level parallelism and thus restricted itself to leveraging the CPUs of the HPC platform. A high-compute node of the Summit supercomputer consists of 2 IBM POWER9 CPUs with 42 cores and 6 Nvidia V100 GPUs.

4. EXPERIMENTAL RESULTS

The performance of the parallel implementation using Dask was compared with the previously developed serial implementation deployed on a high-compute node of Summit using 1 CPU core as a baseline as depicted in Table 1. The serial implementation took 4974.91 seconds to complete vector data alignment with the raster remote sensing imagery, while the parallel implementation took 3037.03 seconds; thus achieving a speedup factor of 1.63.

Table 1. CPU-level Parallel and Serial execution runs of Automatic Vector Data Alignment deployed on Summit supercomputer

	Execution Times (in seconds)		
HPC Platform	Parallel	Serial	Speed Up
Summit	3037.03	4974.91	1.63

To further demonstrate the scalability of the parallel implementation towards efficient resource utilization, experiments were performed by consistently increasing the number of CPU cores used in the high-compute node of Summit. Figure 3 presents the graph documenting the execution times of the parallel implementation as the number of CPU cores are consistently increased. It can observed that execution time is almost linear for the initial runs up-to 16 cores, and then the execution time improves as the number of cores are increased. This performance improvement is largely attributed to Dask's default thread scheduler that scales the task graph execution over the available number of CPU cores.

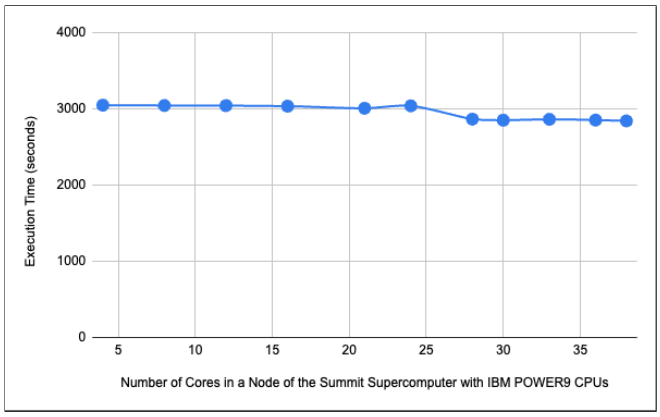

Fig. 3. Graph depicting performance impact on increasing the number of CPU cores for parallel execution using Dask for the Automatic Vector Data Alignment workflow deployed on one node of the Summit Supercomputer

5. CONCLUSION AND FUTURE WORK

In the background of massive volumes of geospatial data generated by remote sensing platforms, the need for scalable GeoAI workflows has become increasingly evident. In context to disaster preparedness and response activities and humanitarian applications where timely response is of

essence, the issue of spatial misalignment between the previously derived vector data and the timely acquired remote sensing imagery, is a major concern. Although there have been research studies focused on automatic geospatial vector data alignment, there exists a research gap in regards to addressing the scalability aspect of these workflows.

This research is aimed towards scaling an automatic vector data alignment workflow. In that regard, this paper proposed to leverage parallel computing in tandem with deployment over a high-performance computing platform to accelerate an automatic vector data alignment workflow. In that regard, this study reported on the preliminary experiments focused on CPU-level multi-core parallelism deployed on a high-compute node of the Summit supercomputer. Considering the improved performance of the parallel implementation as compared to its serial counterpart, leveraging Dask for scaling geospatial workflows seems a promising path ahead. This also inspires adaption of similar scaling strategies for maximizing the throughput of pre-processing and post-processing tasks promoting an integrated GeoAI workflow deployable on high-performance computing platforms.

The future work of this research would involve experiments for further improving the parallel implementation by exploring *Dask Futures* and *Dask Distributed* towards leveraging multi-node multi-GPU level parallelism on high performance computing platforms. Tasking multiple GPUs in a node as schedulable resources would further enable effective resource utilization for the existing GeoAI workflows. With multi-node multi-GPU cluster to be deployed on high performance computing platforms, future scalability experiments would also involve measuring weak scaling governed by Gustafson's law[11] and strong scaling governed by Amdahl's law[12] in addition to benchmarking other geospatial vector data based workflows. The accelerated and scalable GeoAI workflows envisaged would enable and help achieve the capability to cater for rapid response to time-critical requests and further the research for planet-scale geospatial analysis.

6. ACKNOWLEDGMENT

This research used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.

7. REFERENCES

[1] J.McKee and M.Laverdiere, "Automated registration of vector data to overhead imagery," in *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*, 2021, pp. 5465–5468.

[2] T.Liu and D.Lunga, "Automated openstreetmap data alignment for road network mapping," in *IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium*, 2020, pp. 2603–2606.

[3] Y.Tanguy, J.Michel, and G.Salgues, "Automatic Registration of Vector Data with Optical Images," *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 43B4, pp. 191–196, Aug. 2020.

[4] W.Duan, Y.-Y.Chiang, C. A.Knoblock, V.Jain, D.Feldman, et al., "Automatic alignment of geographic features in contemporary vector data and historical maps," in *Proceedings of the 1st Workshop on Artificial Intelligence and Deep Learning for Geographic Knowledge Discovery*, Los Angeles California, Nov. 2017, pp. 45–54, ACM.

[5] W.Duan, Y.-Y.Chiang, S.Leyk, J. H.Uhl, and C. A.Knoblock, "Automatic alignment of contemporary vector data and georeferenced historical maps using reinforcement learning," *International Journal of Geographical Information Science*, vol. 34, no. 4, pp. 824–849, Apr. 2020.

[6] C.Tomasi and R.Manduchi, "Bilateral filtering for gray and color images," in *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, 1998, pp. 839–846.

[7] J.Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986.

[8] Z.Li, "Geospatial big data handling with high performance computing: Current approaches and future directions," *High Performance Computing for Geospatial Applications*, pp. 53–76, 2020.

[9] "Dask — Scale the Python tools you love — dask.org," https://www.dask.org/, [Accessed 25-May-2023].

[10] Microsoft, "US Building Footprints," https://github.com/microsoft/USBuildingFootprints, [Accessed 25-May-2023].

[11] K.Moreland and R.Oldfield, "Formal metrics for large-scale parallel performance," in *High Performance Computing: 30th International Conference, ISC High Performance 2015, Frankfurt, Germany, July 12-16, 2015, Proceedings 30*. Springer, 2015, pp. 488–496.

[12] M. D.Hill and M. R.Marty, "Amdahl's law in the multicore era," *Computer*, vol. 41, no. 7, pp. 33–38, 2008.