FEATURE EXTRACTION AND TRACKING FOR LARGE-SCALE GEOSPATIAL DATA

Lina Yu, Feiyu Zhu, Hongfeng Yu, Jun Wang

University of Nebraska-Lincoln
Lincoln, NE 68588, USA

Kwo-Sen Kuo

University of Maryland
College Park, MD 20740, USA

ABSTRACT

Feature extraction and tracking is a fundamental operation used in many geoscience applications. In this paper, we present a scalable method for computing and tracking features on distributed memory machines for large-scale geospatial data. We carefully apply new communication schemes to minimize the data exchanged among the computing nodes in building and updating the global connectivity information of features. We present a theoretical complexity analysis, and show that our method can significantly reduce the communication cost compared to the traditional method.

Index Terms— Feature extraction and tracking, large-scale data, geospatial data, parallel and distributed computing

1. INTRODUCTION

Steady advance in remote sensing, satellite imaging, and computing technology has enabled scientists to collect geophysical phenomena data with unprecedented resolutions and complexity. Feature extraction and tracking is one fundamental analysis task performed on Earth observation datasets, where scientists wish to select and track regions of interest (RoIs) by querying either the primary variables or complicated functions of the primary variables, and possibly assemble the selected RoIs into a meaningful time series. In this way, scientists may address important questions determining what conditions are normal, and how those conditions (e.g., climate) might be changing over time. For example, Figure 1 shows a map of Ertel's potential vorticity for a day in January 2013 that renders marginal evidence of a strong Northern Hemisphere polar vortex. Scientists want to extract and track this vortical structure to gain a detailed examination of other variables (e.g., zonal mean temperature and winds) in the corresponding RoIs for a verification of their hypothesis.

However, it becomes increasingly challenging to identify and track features from Earth observation data. A dataset generated from space-based satellites or ground-based radar and radiometer facilities is typically time-varying, and multivariate, and can take tera- or even peta-bytes of space to preserve and process. To tackle the vast amounts of data, distributed computing provides a scalable and feasible solution,

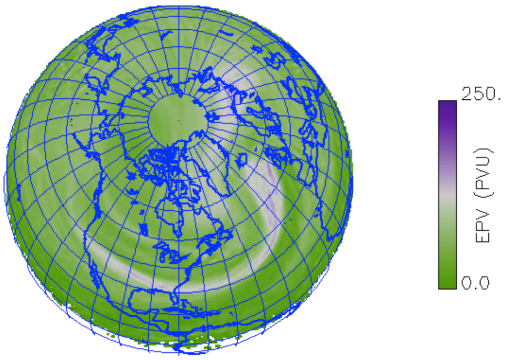

Fig. 1: A polar vortex (the purple area) marked by high Ertels potential vorticity (EPV) on the 600 K potential temperature surface during a stratospheric sudden warming.

where the dataset is partitioned and distributed among multiple computing nodes, each computing node identifies partial components of a feature from its local data block, and then a complete feature is assembled from partial features. However, most existing feature extraction and tracking algorithms [1, 2, 3, 4] have focused on how to quantity features based on different measures, such as size, location, shape, or topology information. These algorithms are effective to extract partial features from an individual data blocks, but do not provide a scalable mechanism to assemble partial features to form an integrated description of a complete feature.

2. BACKGROUND

The key of distributed feature extraction and tracking is twofold. First, we need to build the connectivity information of a feature across multiple computing nodes. Such information facilitates us to obtain a global description of partial features from a set of distributed nodes, and enables subsequent operations, such as querying the values of other variables within the region of the feature. Second, we need to update the connectivity information of features that can evolve over time.

Figure 2 illustrates an example of the evolution of a vortical structure, f_m. We assume the dataset is partitioned and distributed among four computing nodes, N_1 - N_4. At a time

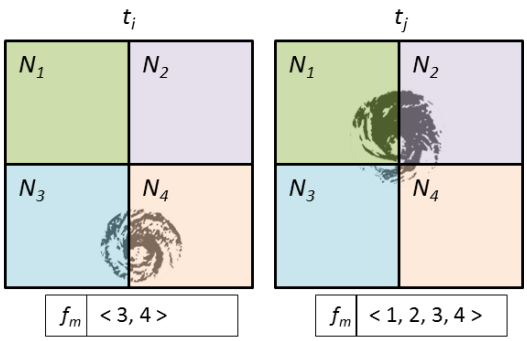

Fig. 2: A vortical feature f_m and its connectivity information dynamically evolve over two time steps, t_i and t_j.

step t_i, the feature is detected on N_3 and N_4, and thus a global connectivity description of this feature is $< 3, 4 >$. At a subsequent time step t_j, the feature evolves and is identified on all four nodes, and thus the description of this feature becomes $< 1, 2, 3, 4 >$. The connectivity description makes it easy for us to identify the host nodes for a certain feature.

However, it is non-trivial to generate and maintain such connectivity information in a distributed environment. A straight-forward method is to use the master-slave paradigm in that each slave node first extracts its local partial features and then sends the information to a master node to generate the aggregated result. For u nodes and v features, the total communication cost is $O(uv)$, which can cause severe link contention with respect to an increasing number of features and computing nodes.

Only a few research work has been carried out to address this issue. Wang et al. [5] proposed a decentralized approach for extracting and tracking features from large-scale 3D volumetric datasets. In their method, each computing node first constructs a partial table of connective information, and then iteratively communicates with its immediate neighbors to exchange and propagate the feature connectivity information. This procedure mimics region growing that starts from one seed and grows to adjacent regions in a breadth-first fashion until all features are connected. Given the 3D data partitioning and distribution scheme in their applications, the number of nodes involved in each communication is less than six.

3. OUR APPROACH

We extend Wang's approach [5] from 3D volumetric datasets to Earth observation datasets. Figure 3 illustrates the major steps of our approach.

3.1. Data Partitioning and Distribution

We regularly partition a data on the spherical surface into a set of patches along the latitude and longitude lines, as shown in

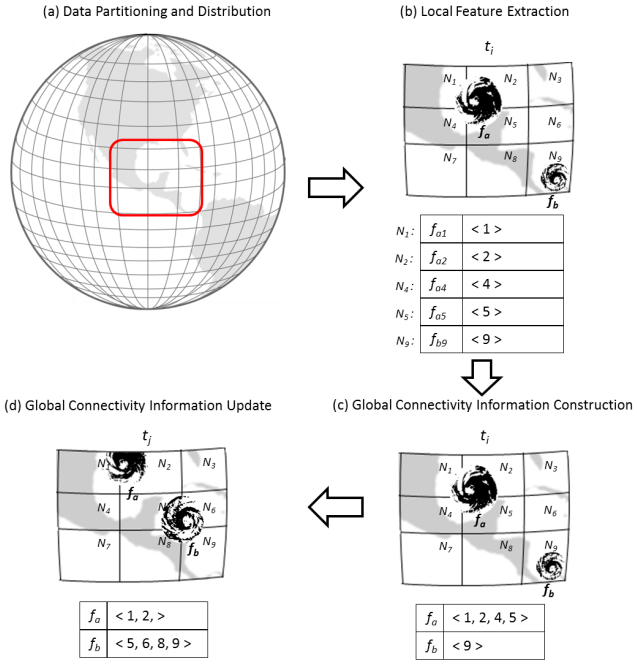

Fig. 3: The major steps of our approach.

Figure 3(a), and then distribute the patches among the computing nodes. For example, given the patches of an observation data within the red circled region in Figure 3(a), we assign them among nine nodes N_1 - N_9. Each patch has an approximately same size such that the workload of feature detection will be balanced among the computing nodes.

3.2. Local Feature Extraction

Each computing node attempts to extract features from its local patch according to user specified feature definitions. Various standard techniques, such as region growing and iso-surfacing, can be used to identify local features. Figure 3(b) shows an example of two features across the computing nodes. We can see that the feature f_a has been identified by the nodes N_1, N_2, N_4, and N_5, and the feature f_b has been identified by the node N_9. Initially, each node does not recognize the connectivity of their local partial features, and constructs a partial feature table for the local partial features with their local IDs, such as f_{a_1}, f_{b_9}, and so on. Correspondingly, the connectivity description of a local partial feature only contains the host node ID. Figure 3(b) shows the structure of partial feature table on each node that contains local features.

3.3. Global Connectivity Information Construction

We then use a two-passes communication scheme to build global connectivity information. In the first pass, each node

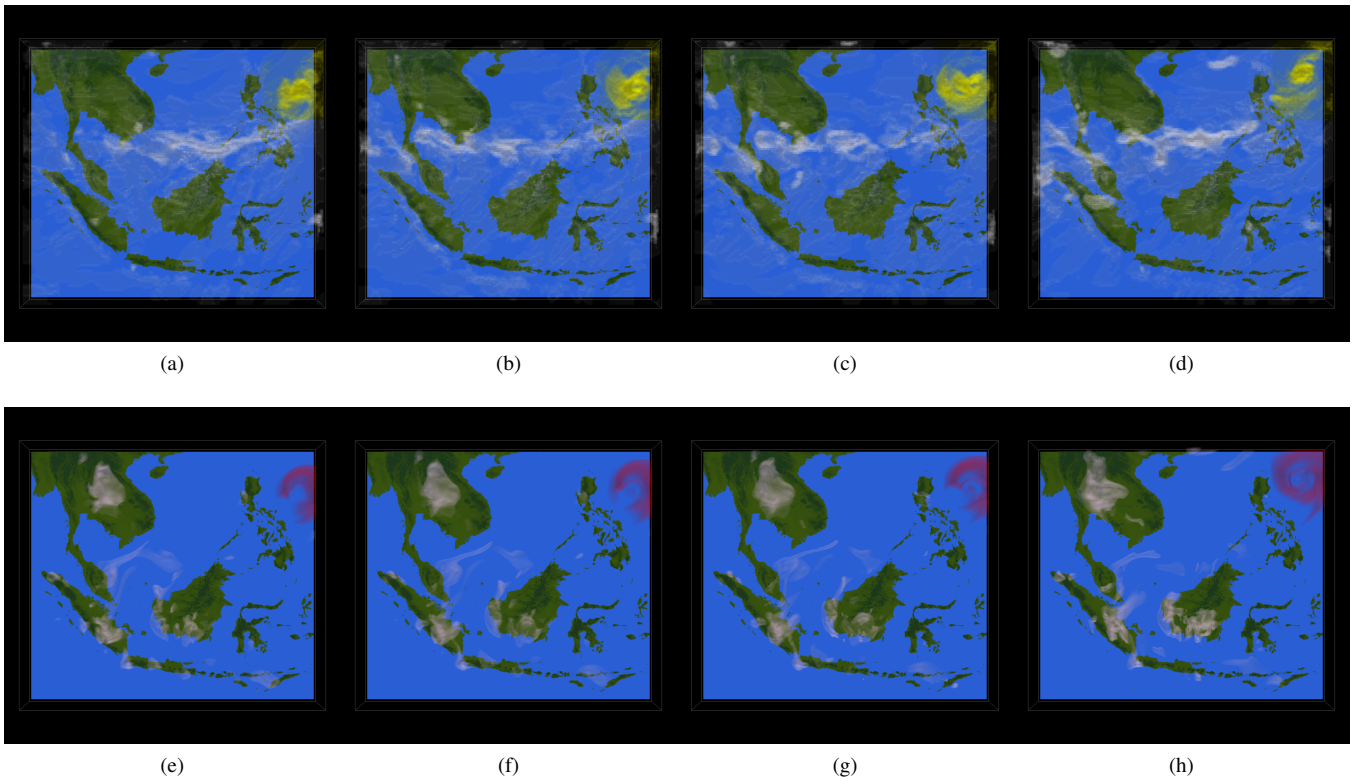

Fig. 4: Visualization of the interplay between wind and smoke pathway over time. (a)-(d): we extract and trace the core region (in yellow) of a typhoon, where the white and yellow regions have similar high $CLDFRA$ values. (e)-(h): we extract and trace the aerosol mass (in red) trapped by the typhoon, where the pink and red regions have similar high $PM10$ values.

exchanges its local table with its direct neighbors along the latitude direction. For example, after exchange, N_1 has two tables: f_{a_1}:$< 1 >$ and f_{a_2}:$< 2 >$. By comparing the boundary information of the partial features, N_1 can easily recognize that f_{a_1} and f_{a_2} correspond to the same feature, and thus merge two tables into one: $f_{a_{12}}$:$< 1, 2 >$. Similarly, N_2 has the table $f_{a_{12}}$:$< 1, 2 >$, N_4 and N_5 have the table $f_{a_{45}}$:$< 4, 5 >$, and N_9 has the table f_{b_9}:$< 9 >$.

In the second pass, each node exchanges its local table with its direct neighbors along the longitude direction. Then, N_1 has two tables: $f_{a_{12}}$:$< 1, 2 >$ and $f_{a_{45}}$:$< 4, 5 >$. By comparing the boundary information of the partial features, N_1 can also recognize that $f_{a_{12}}$ and $f_{a_{45}}$ belong to the same feature, and then generate a merged table $f_{a_{1245}}$:$< 1, 2, 4, 5 >$. We can easily see that N_2, N_4, and N_5 can generate the same table as N_1. N_9 still has the table f_{b_9}:$< 9 >$.

We repeat this two-passes communication scheme until the table of each node becomes stable. The number of repetition depends on the size of features. For a larger feature that covers more nodes, it requires more communication to propagate the partial tables among the related nodes. This situation, however, is rare in practice.

Once there is not change in its local table, each node gets the global connectivity information of a feature, and assigns a unified global ID to this feature. For example, $f_{a_{1245}}$ is changed to f_a, and f_{b_9} is changed to f_b, as shown in Figure 3(c).

3.4. Global Connectivity Information Update

To track features at a new time step t_j, we still first extract local features at each node, and then use a similar multiple-pass communication scheme to update the global connectivity information of a feature. As shown in Figure 3(d), we can see that f_a has partially moved outside the domain, and its table shrinks to f_a:$< 1, 2 >$. On the other hand, f_b has been enlarged, and its table becomes f_b:$< 5, 6, 8, 9 >$.

4. RESULT

We apply our approach to assist atmospheric scientists in visualizing mesoscale modeling of smoke transport over the Southeast Asian Maritime Continent [6]. In this area, dry conditions associated with the moderate El Niño event leaded to the largest regional biomass burning outbreak since 1997. The 3D distribution of smoke particles highly manifests the

complexity of meteorology in the Martine Continent, especially for the interplay of sea/land breeze, typhoons, and storms over the subtropical western Pacific Ocean, as well as topographic effect and trade winds that can be clearly seen in the model simulation of smoke transport. Smoke and burning emission is specified according to the location with high values of $PM10$ (particulate matter with diameter less than 10 μm). The modeled smoke transport pathway can be found by the visualization of the mass concentration of $PM10$. Wind patterns can be conveyed by the dynamics of $CLDFRA$ (cloud fraction).

Figure 4 shows the result for this simulation data using our approach. We first extract and trace the typhoon's center as shown in Figure 4 (a)-(d). The white and yellow regions have high $CLDFRA$ values. Our approach can extract the core region of the typhoon and trace it over time. We also identify the region with high $PM10$ around the typhoon's center, as shown in Figure 4 (e)-(h). Note that the pink and red regions have similar $PM10$ values, and our approach can successfully separate them and trace the relevant region (in red) over time. The result can clearly convey the wind patterns and the movement of aerosol mass, and help scientists study the interplay between them.

5. DISCUSSION

We discuss the complexity of our approach:

- **Local feature extraction time**: Because we use standard algorithms (such as region growing) to extract features, the computation time is determined by the size of data and the number of computing nodes. As we maintain a similar size of each patch, the computation time for feature extraction at each node is approximately the same. When we use more nodes, the size of each patch decreases, and thus does the time spent on local feature extraction.

- **Global connectivity information construction time**: As we choose the four-direct-neighbor communication paradigm, the communication cost is minimized. Given u nodes, the complexity of communication is bounded by $O(\sqrt[2]{u})$, which corresponds to the maximum communications needed for propagating the information of a feature covers the whole domain. The number of nodes involved in each communication is a constant of a maximum of two.

- **Global connectivity information update time**: As we use a similar communication paradigm to update global connectivity information, the complexity of this step is similar to the construction of global connectivity information.

6. CONCLUSION

In this paper, we present a distributed feature extraction and tracking approach for large geospatial data. By carefully applying new communication schemes, we can minimize the data exchanged among computing nodes and prevent link contention. Compared to the traditional master-slave paradigm, our approach can significantly reduce communication cost, achieve balanced workload, and ensure the scalability over a large number of computing nodes. We will conduct a detailed experimental study on large machines to demonstrate the efficiency of our approach.

7. ACKNOWLEDGEMENT

This research has been sponsored in part by the National Science Foundation through grants IIS-1423487 and ICER-1541043, and the NASA Advanced Information Systems Technology (AIST) program.

8. REFERENCES

[1] E. Mesrobian, R. R. Muntz, J. R. Santos, E. C. Shek, C. R. Mechoso, J. D. Farrara, and P. Stolorz, "Extracting spatio-temporal patterns from geoscience datasets," in *Visualization and Machine Vision, 1994. Proceedings., IEEE Workshop on*. IEEE, 1994, pp. 92–103.

[2] C.-R. Shyu, M. Klaric, G. J. Scott, A. S. Barb, C. H. Davis, and K. Palaniappan, "GeoIRIS: Geospatial information retrieval and indexing system–content mining, semantics modeling, and complex queries," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 45, no. 4, pp. 839–852, 2007.

[3] D. Silver and X. Wang, "Tracking and visualizing turbulent 3d features," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 3, no. 2, pp. 129–141, 1997.

[4] G. Ji, H.-W. Shen, and R. Wenger, "Volume tracking using higher dimensional isosurfacing," in *Visualization, 2003. VIS 2003. IEEE*, Oct 2003, pp. 209–216.

[5] Y. Wang, H. Yu, and K.-L. Ma, "Scalable parallel feature extraction and tracking for large time-varying 3D volume data," in *Proceedings of the 13th Eurographics Symposium on Parallel Graphics and Visualization*, Aire-la-Ville, Switzerland, Switzerland, 2013, EGPGV '13, pp. 17–24, Eurographics Association.

[6] C. Ge, J. Wang, and J. S. Reid, "Mesoscale modeling of smoke transport over the southeast asian maritime continent: coupling of smoke direct radiative effect below and above the low-level clouds," *Atmospheric Chemistry and Physics*, vol. 14, no. 1, pp. 159–174, 2014.