THE ROLE OF COMPRESSION IN SPATIAL COMPUTING

G. Dax[1], M. Werner[1]

[1] Technical University of Munich, Germany; School of Engineering and Design, Department of Aerospace and Geodesy, Professorship for Big Geospatial Data Management - gabriel.dax@tum.de, martin.werner@tum.de

KEY WORDS: Compression, Bloom filter, Deep Learning, Spatial Computing

1. INTRODUCTION

Spatial computing is a field in which algorithms from computational geometry, noisy data from sensors, and methodology from data science including data mining, machine learning, and artificial intelligence melt into a domain focusing on how to solve the computational challenges occurring especially with space-time data. In this field, numeric accuracy has been a traditional aim due to many visual and algorithmic artifacts occurring from rounding errors in spatial data analysis. As a consequence, the contributing data acquisition technologies (geodetic measurements, GNSS, RADAR) have been traditionally evaluated not on a certain application, but rather with measures of data quality not related to end-to-end applications.

In the last decades, especially with rising interest in deploying end-to-end machine learning methodology to the field, however, this picture is changing. Fitness for purpose in various applications replaces abstract quality metrics and data simplification (e.g., downsampling of images, simplification of geometries) becomes a common part of the processing pipeline.

These observations make a question obvious: when designing, for example, a satellite system for a certain application, which sensor quality do we really need? How much downstream capacity for sending data over a wireless link from space to Earth do we want to invest? Can we work with a cheaper or more energy-efficient processing system? From a more general perspective, the question is about information compression in a given application. How much of the input information and fidelity (resolution, bits, sampling) is needed and how much of the information can be compressed early in the data flow?

Following the philosophy of Kolmogorov, we realize that the information complexity of a data object can be defined as the length of the shortest program that generates the information. That is, data, parameters, and programs are to be treated the same. In relation to this thesis project, therefore, we need to ask:

1. How can we compress input data
2. How can we compress program parameters (e.g., weights in deep learning)
3. How can we compress intra-program communication (e.g., candidate sets, intermediate results)
4. How can we compress the output
5. How can we compress the program itself
6. How can we compress the hardware itself

Note that compression here is not a compression algorithm, but any mechanism of thought that makes the thing smaller. This includes simple solutions such as reducing the number of parameters, optimizing code generation for space occupancy (compressing the program), or for the highest performance (usually increasing the size of the program).

2. RESEARCH ACTIVITIES

Currently, this research is decomposed into three lines of innovation, namely, how to get deep learning computer visions run in energy, resource, and space-constrained environments like in Earth observation satellites, how to reduce the data communication by first constructing descriptors of geometric objects that are plain sets, and how to make use of compressed data directly using Kolmogorov inspired distance notions.

2.1 Running Deep Learning on Space- and Energy Constrained Hardware in Space

In a project between European Space Agency (ESA), a startup building forest fire detection satellites (OroraTech), a major space manufacturer (Airbus), and hardware experts from the Technical University of Munich, we are exposed to a huge set of claims from the remote sensing community what one could do with deep learning in the lab (e.g., cloud detection, fire detection, surface water detection, land cover land use, etc.) contrasted with extremely limited processing abilities (radiation, energy-budget, communication channel to Earth) in reality. We are developing a benchmarking platform showing the relations between machine learning performance measures (e.g., precision, recall, F1), energy performance (e.g., mW/image), frame rate (e.g., frames per second), and computing device (e.g., FPGA space utilization).

This project aims to provide an ESA-encouraged benchmarking environment steering the community of remote sensing scientists back to realistic assumptions about the available computation.

2.2 Turning Attention from Numeric Features (tables) to Binary Features (e.g., sets)

In a second stream of work, we consider that the removal of any order structure from our data provides a sketch of the data that can be handled much simpler, maybe even without understanding the data at all just measuring information, for example, by entropy. A framework was developed in which geometric objects like spatial trajectories are observed, transformed into ordered sequences. Then, the order sequence is explicitly forgotten and it is surprising to see how much similar information can still be extracted, for example, using methodology related to text mining.

2.3 Using compressed data directly

In general, it is very common that in all systems that employ compression as part of the data pipeline, decompression operations are inserted. Unfortunately, these operations can be extremely costly especially in random access scenarios as most real-world compressors compress larger blocks of data, and looking into such a block essentially implies a complete decompression of the block. Bloom filters have been used quite a lot to avoid these decompressions (or external reads in distributed databases), but they are limited to situations in which set-like sketches can be generated easily (including, but not limited to key-value stores).

In data mining, it is less common to directly work with compressed data even though this is possible. The Kolmogorov distance between two abstract data objects can be introduced based on the notion of the shortest program generating the two data objects in one program compared to the length of a program generating the first data object and a different program independently generating the second data object. This idea generates a proper distance with all properties one would expect, but unfortunately, cannot be computed. One can now approximate this distance by replacing the Kolmogorov complexity with the length of a real-world compression of the data objects. That is, one compresses data object A and B, for example, with ZIP and as well the concatenation of the data objects A and B.

This approach works very well (Dumitru et al., 2019), but the distance computation is far from trivial as for an unknown query object q, not only a compression of q, but as well the joint compression of q with all database items a_i needs to be computed. A key observation of this thesis is that when interpreting Bloom filters as compressions of sets, we are provided with constant-time access to the joint compression boiling down to a binary OR between the query filter and the database. We are currently exploring the implications of this and working on solving the open challenge, of how suitable geometric information can be captured in terms of a set. Initial work on raster images (Werner, 2019, Werner, 2021) and spatial trajectories (Dax and Werner, 2021) has been done, point cloud data is under active investigation.

3. CONCLUSION

With this thesis, we want to understand compression in spatial data science including both the computational dimension of compression (faster algorithms, smaller hardware, smaller program code) and the data dimension of compression (less input, less intermediate information, less output). With strong links to real-world applications (AI in space) and computer science (Kolmogorov information theory, compression distance), we hope to make a significant contribution to spatial computing unlocking the perspective that fewer data and computation can sometimes lead to the same or even better results.

REFERENCES

Dax, G., Werner, M., 2021. Trajectory similarity using compression. *2021 22nd IEEE International Conference on Mobile Data Management (MDM)*, 169–174.

Dumitru, C. O., Dax, G., Schwarz, G., Cazacu, C., Adamescu, M. C., Datcu, M., 2019. Accurate monitoring of the danube delta dynamics using copernicus data. *SPIE Remote Sensing*, 1–13.

Werner, M., 2019. Globimaps - a probabilistic data structure for in-memory processing of global raster datasets. *27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (SIGSPATIAL'19)*.

Werner, M., 2021. GloBiMapsAI: An AI-Enhanced Probabilistic Data Structure for Global Raster Datasets. 7(4). https://doi.org/10.1145/3453184.