Deep Learning based Approach for Fast, Effective Visualization of Voluminous Gridded Spatial Observations

Paahuni Khandelwal, Menuka Warushavithana, Sangmi Lee Pallickara, Shrideep Pallickara
Department of Computer Science, Colorado State University
Fort Collins, Colorado, 80521
paahuni@colostate.edu, menukaw@colostate.edu, sangmi.pallickara@colostate.edu, shrideep.pallickara@colostate.edu

***Abstract*—Gridded spatial datasets arise naturally in environmental, climatic, meteorological, and ecological settings. Each grid point encapsulates a vector of variables representing different measures of interest. Gridded datasets tend to be voluminous since they encapsulate observations for long timescales. Visualizing such datasets poses significant challenges stemming from the need to preserve interactivity, manage I/O overheads, and cope with data volumes. Here we present our methodology to significantly alleviate I/O requirements by leveraging deep neural network-based models.**

I. Introduction

The proliferation of sensors, observational equipment, and networked measurement devices has contributed to substantial growth in data volumes. In addition to the observations of interest, spatial datasets have geocodes (such as $\langle lat/long \rangle$), associated with them. Several data also have timestamps associated with the observations; the observations are often multidimensional encapsulating multiple, related features of interest. We consider gridded spatial datasets that frequently arise in climate, meteorological, ecological, and satellite-based remote sensing datasets. In gridded spatial datasets, the data are available at fixed, spatially dispersed points based on the spatial resolution which determines the spatial increments at which data are available.

Scientists, stakeholders, and users alike rely on visualizations to understand spatial variation of phenomena. The pairwise combinations in which they can be layered have the asymptotic bound of $O(N^2)$ where N is the number of variables. Each feature of interest may be visualized independently, or multiple features may be layered to understand how they vary with respect to each other. The crux of this effort is to enable interactive visualizations of gridded spatial datasets.

The paper is an extended abstract of [1] and included as part of the Early Career and Students' Showcase.

II. Approach Summary

Our methodology targets interactive visualizations of gridded spatial datasets. We design and train models that generate effective visualizations. Rather than extensively performing disk and network I/O during visualizations, we use models to render phenomena. We train models for multiple zoom levels while allowing users to interactively engage with the visualizations using panning, zoom-in, and zoom-out operations.

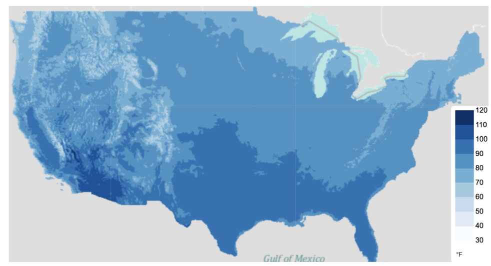

Fig. 1: Rending maximum air temperature phenomenon over CONUS exhaustively takes up to 56.1805 secs.

Constructing models at different resolutions that are aligned with zoom levels allows us to reduce model complexity while ensuring fidelity.

Rather than exhaustively retrieve all data that must be visualized, we rely on retrieving a fraction of the dataset; this fractional dataset is then used by our models to render phenomena while ensuring fidelity and preserving interactivity. Inferences are performed during the critical path of visualizations. During inferences, we seed the model(s) responsible for rendering tiles within a viewport with a fraction of the ground truth data.

Our benchmarks demonstrate the suitability of our approach to rendering visualizations. Consider exhaustively rendering phenomena by retrieving all observations; this takes 56.1805 secs to complete the fetch-and-render operations. Our approach of seeding machine learning models renders the same phenomenon in 4.3043 secs, a 92.3% reduction in rendering times while preserving a high PSNR accuracy of 38.7 dB.

The proposed methodology does not make any assumptions about the underlying spatial referencing system. As such, it is broadly applicable to gridded datasets that arise in other domains such as computational fluid dynamics. Similarly, gridded datasets occur in non-terrestrial settings such as atmospheric and oceanic phenomena; this work translates to those as well.

III. Methodology

Our methodology targets the visualization of gridded datasets; each grid point is identified by $\langle lat/long \rangle$ coordinates and includes a vector of observations alongside a timestamp. In this study, we consider one of the most well-known

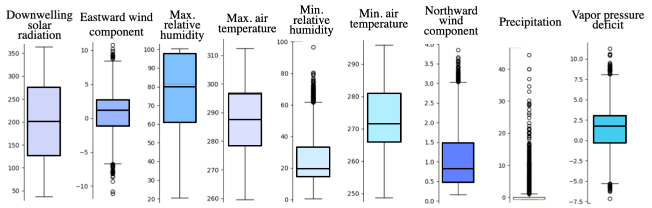

Fig. 2: Range of values for various meteorological parameters at sampled locations from MACA dataset.

gridded datasets, MACA Multivariate Adaptive Constructed Analogs) [2]. The MACA dataset represents an amalgamation of over 20 global climate models (GCMs) downscaled to 4km (1/24th degree) spatial resolution and representing different outcomes for the Radiative Concentration pathways for greenhouse gas emissions. We consider the RCP 8.5 trajectory. We consider data encompassing projections for the years 2023-2030 for the continental United States (CONUS).

Figure 2 shows the meteorological parameters of the dataset that we focused on and the range of their values. This includes multiple parameters such as maximum and minimum temperature, maximum and minimum relative humidity, precipitation accumulation, downward surface shortwave radiation, wind velocity, and specific humidity.

Our methodology involves seeding the models with a fraction of the actual data. This allows us to alleviate expensive disk and network I/O requirements by leveraging DNN models to render the phenomena. Rather than using the entire set of available observations, we train DNN models to super-resolve and learn non-linear interactions from a small fraction of available data to render the finest resolution tile of size 64x64 pixels for 8 meteorological parameters.

Our DNN network ingests a fraction of data to infer real-time meteorological information by leveraging self-supervised learning. The core of our deep neural network architecture is inspired by SRGAN [3], which is a super-resolution generative adversarial network as depicted in Figure 3. We provide the model with a location hint by generating an embedding vector for the geostring associated with the input tile. The model also ingests the temporal hint provided as an embedding of the associated month. This helps in learning interactions between the labeled pairs - input and full output images, based on a particular region and seasonality. Both the spatial and temporal hints are passed through the repeat vector layer and reshaped into 4x4x1 dimensions so that they can be merged with low-resolution input data (4x4x10).

The low-resolution input image is generated by selecting every 16th pixel of the tile in both the x and y dimensions for each meteorological parameter. Training the DNN model with limited true information results in very low network I/O performed to get these values from our back-end to the client's device. The model learns to map as low as 0.003906 fractions of true values (16 points) to generate high-resolution visualization of the tile (4096 points). We also allow the single DNN model to train across multiple features of interest by leveraging the non-linear interactions occurring between

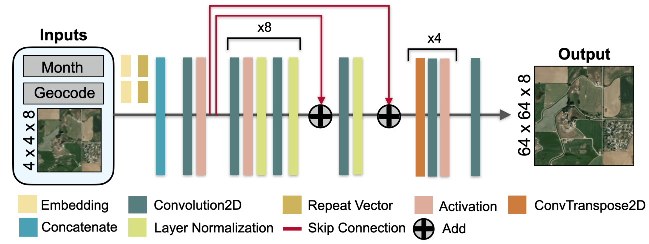

Fig. 3: The architecture for our super-resolution DNN. Repeated cells of convolutional layers with skip connections followed downscaling blocks to increase spatial resolution.

these variables. The meteorological variables are often interconnected; for example, the temperature is inversely proportional to relative humidity throughout the day. Providing the model with all parameters allows the model to learn inter-feature relationships. This low-resolution input image is concatenated with temporal and spatial hints and passed through pre-residual blocks comprising 2D convolution layers and a ReLU activation function. The input meteorological parameters are normalized and scaled individually between 0-1 for land regions. The ReLU activation function ensures that values emerging from the convolution layer do not saturate.

The pre-residual blocks are followed by 8 blocks of residual blocks connected through skip-connections to avoid vanishing gradients. The residual block comprises two convolution layers, layer normalization, and a ReLU activation function. The number of features in convolution layers is increased and kernel size is kept at 5x5 to extract low-level feature maps across the neighboring spatial region in the image in order to retain crucial information for higher spatial resolutions. Here, we perform layer normalization that normalizes the activations along the parameter/feature dimension instead of normalizing the batches. This is to account for the fact that each of the parameter values is at different ranges. Next, we perform upsampling of the image by consecutively increasing the spatial dimension of the features maps using a block of convolutional transpose layers and an activation layer to infer the full image with all 8 output parameters.

The number of output features at each convolutional, convolution layer, learning rates, and kernel sizes is fixed by performing hyper-parameter tuning using Hyperband [4]. Determining a robust set of hyperparameters is crucial for expedited model training and better performance. We leverage the Tensorflow Hyperband tuner which speeds up extensive parameter grid searches through adaptive resource allocation and early stopping to identify best-performing combinations.

Our benchmarks demonstrate that deploying our lightweight models can reduce the client's query response time by 92.3% while maintaining a high perceptual quality with a PSNR (peak signal-to-noise ratio) of 38.7 dB.

IV. Acknowledgement

This research was supported by the National Science Foundation [OAC-1931363, ACI-1553685], the National Institute of Food and Agriculture [COL0-FACT-2019], and a Jack Cochran Family Professorship.

References

[1] P. Khandelwal, M. Warushavithana, S. L. Pallickara, and S. Pallickara, "Enabling fast, effective visualization of voluminous gridded spatial datasets," in *2023 IEEE/ACM 23rd International Symposium on Cluster, Cloud and Internet Computing (CCGrid)*. IEEE, 2023.

[2] C. Lab. Multivariate adaptive constructed analogs (maca). [Online]. Available: https://www.climatologylab.org/maca.html

[3] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4681–4690.

[4] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar, "Hyperband: A novel bandit-based approach to hyperparameter optimization," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 6765–6816, 2017.